Beginner's Performance with MessagEase and QWERTY

Anju Thapa

University of Tampere
School of Information Sciences
Interactive Technology, M.Sc thesis
Supervisor: Poika Isokoski
June 2013

University of Tampere
School of Information Sciences
Interactive Technology
Anju Thapa
M.Sc. thesis, 52 pages, and 6 appendix pages
June 2013

With the increased use of mobile phones, interest in text entry with them has also amplified. Many new mobile phones are equipped with a QWERTY keypad; new methods to surpass the QWERTY performance are also being developed.

This thesis compares user performance of virtual QWERTY keypad to MessagEase. MessagEase uses 9 keys and can therefore be used even on very small touch displays. 9 characters are entered with tapping and the rest with a tap-and-slide gesture.

An experiment was conducted with 10 participants transcribing text with both text entry techniques. The experiment consisted of three sessions. In each session, the participants transcribed 30 phrases in total - 15 phrases using each text entry technique. Responses to the System Usability Scale (SUS) for each text entry technique and informal interview data were also collected.

From a Repeated-measures analysis of variance a significant effect of the text entry method on text entry rate was seen ($F_{1,19}$= 47.140, $p < 0.0001$). The effect of the session (i.e. learning) was also statistically significant ($F_{2,18}$= 3.631, $p = 0.047$).The interaction of the session and method was also statistically significant ($F_{2,18}$= 10.286, $p = 0.001$) indicating different learning rates. Average text entry speed with MessagEase was 7.43 words per minute (wpm) in the first session and 10.96 wpm in the third session. Likewise, text entry speed with the QWERTY soft keyboard was 17.75 wpm in the first session and 17.16 wpm in the third session. No significant difference was found in the error rates.

Key words and terms: text entry method, MessagEase, QWERTY.

Table of Contents

Acknowledgments

Firstly, I would like to thank my thesis supervisor, Prof. Poika Isokoski for his tireless and continuous support in completing my thesis. I would also like to thank all the participants who volunteered for the experiment. My sincere thanks to all my friends, and especially, Kendra, Sumita, Manik and Joel, who supported and guided whenever needed.

A special thanks to my family, especially my mother, Ambika Devi Chhetri and sisters, Anupama and Arati who are the reasons of what I am now. Without their love, care, support and guidance nothing would have been possible.

1 Introduction

Mobile phones are among the most useful developments in recent years. In the 38 years that mobile phones have existed the growth has been immense. At the start of the 21st century, mobile phones are not considered a luxury but a necessity for every person. From the demonstration of the first hand-held phone in 1973 by John F. Mitchell and Dr. Martin Cooper of Motorola (Teixeira, 2010) until the present, mobile phone development has varied a lot with respect to efficiency, portability, aesthetics and functions. From 1G to 3G mobile phones, several new features have been introduced. Making and receiving a call, Short Messaging Service (SMS), camera, high-speed Internet connection, games, alarm, calendar and music are considered to be common features found in almost every newly manufactured mobile phone.

SMS was first started in 1992 with messages sent from computer to mobile phone, and in 1993 from mobile phone to mobile phone (Wikipedia, 2012a), however the concept had already been created in the 1980s in Europe. Although it did not capture the market or customer attention in its initial phase, SMS is one of the world's most used data communication technology, for instance, according to the Guardian report, more than 6 trillion SMS's were sent in 2010 (Naughton, 2012). Texting is an important feature that is used extensively by mobile users.

In general, the act of sending an electronic message from one mobile to another, or between electronic devices, which are capable of communicating through text, is known as texting. Recently, there has been a growing interest in finding efficient text entry techniques. What then makes a new innovation of text entry successful? Certainly, there has been a huge evolution in text entry technique and several factors are considered in any text entry technique. Is it dependent on the simplicity of the keyboard layout or its efficiency? Several questions need to be answered before developing a new technique. With various techniques such as multi-tap, T9, Dictionary prediction on QWERTY keyboard and gestural keypads, developers have tried to pay attention on increasing the effectiveness and speed of text entry. Not all techniques though, are successful in grasping the user's attention in the long run.

It is definitely not easy convincing users to learn new text entry techniques when they are comfortable with the existing ones. For the success of any new innovation, focus has been on comparing the new technique with the existing ones. A detailed study of the factors that could make one method better than the existing ones should also be considered. Comparing the new method with the existing ones also helps to shape new techniques into perfection. Almost all who started using mobile phones in the 80's or 90's are familiar with the multi-tap method. Hence, it is possible that many people do not prefer changing their text entry method as they might find new methods cumbersome to use, although it might get better with practice.

The most widely adopted text entry method in mobile phones is the multi-tap method, which requires users to press key multiple times for entering letter. It remained popular for ages due to its availability on many mobile phone models. Further, to increase the efficiency of text entry and to decrease the number of key taps, different predictive text entry methods such as T9 and LetterWise have been developed. Word prediction reduces the text entry effort; however it was not always able to predict the desired word. In some cases where users wanted to text in their native language or some abbreviations, text prediction failed. Another remarkable advancement in text entry method was the adoption of the 'QWERTY' method of desktop keyboards in mobile keypads. As the layout was familiar, it was able to capture the market soon after its integration in mobile devices. QWERTY removed the problem of multiple taps in a single key; however fitting a QWERTY keyboard on a mobile phone resulted in smaller key size.

To solve this lack of key space for text entry, Nesbat developed MessagEase, a unique way of text entry with only a 9 key arrangement, thus providing greater space to write. With 9 large keys and tap and drag features, MessagEase is designed to maximize text entry speed with fewer errors. However, experimental evidence is yet to be found for analysing the efficiency of MessagEase text entry technique.

Theoretically, MessagEase is believed to be a fast text entry method. We have anecdotal evidence of very fast text entry speed with MessagEase, a world record of 84 WPM (Exideas, 2012a). However, the acceptance of any new text entry techniques also depends on the learning path of the user. A longitudinal study on the performance of menu-augmented soft keyboards by Isokoski (2004) suggests that adding a marking menu would speed up text entry resulting in the higher text entry for expert users, however reaching that expertise level take time. In the study two experiments was conducted using QWERTY layout with and without the use of menu. From the study it was concluded that menu-augmented soft keyboard is faster with the use of QWERTY layout when the participants are expert in the use of menu(Isokoski, 2004). From the evidence we have reasons to suspect that learning tap-stroke text entry is slow. Users may not prefer text entry techniques with fast expert performance but painful and long learning path. Hence, measuring a beginner's performance with MessagEase is critical for its success. In order to measure data for beginner's performance with MessagEase, I conducted an experiment.

The goal of the work I report in this thesis was to compare the performance of MessagEase and QWERTY text entry methods for a beginner. The study focuses on analysing the speed (words per minute) and accuracy of novice MessagEase users to see whether or not it can match the QWERTY performance.

In this thesis, chapter 2 describes key-based text entry methods such as 12-key keypad and QWERTY keypad. Similarly, chapter 3 describes different stylus based text entry methods. sChapter 4 deals with the metrics of text entry evaluation. Chapter 5 describes the methods of the experiment comprising of the study hypothesis, participants, apparatus, test procedure and study design. Results of the experiment are then analyzed in chapter 6. Finally, in chapter 7, results of the experiment are discussed along with the conclusion.

2 Key-based Text Entry

A Key-based text entry technique involves the use of a keyboard in text entry. It could be the physical keyboard in mobile phones, or virtual keyboard in smart phones, where text entry is performed with the help of stylus or finger. Similarly, the keyboard layout could be a full keyboard with 27 or more letters in the keyboard or ambiguous keyboard with two or more letters per key. The arrangement of letters in a keyboard determines the keyboard ambiguity. Keyboard layout with a single letter in each key is less ambiguous and the ambiguity increases with the increase of letters in a single key(see Figure 1).

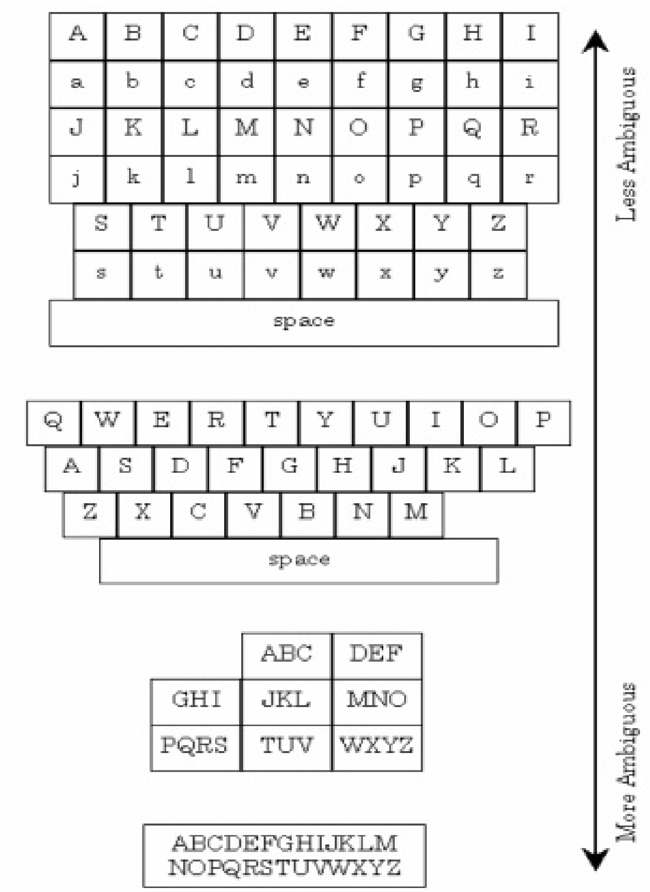

Figure 1: The "key-ambiguity continuum"

(MacKenzie & Soukoreff, 2002)

This chapter briefly summarizes key-based text entry techniques ranged from 12-key keypad to QWERTY layout, Five key text entry, SHK Card, TiltText and Chorded keyboard.

2.1 12-key Keypad

Text entry method in a standard mobile phone is based on a 12-key keypad such as a traditional telephone keypad which consists of ten number keys from 0 to 9 and two additional keys * and # (see Figure 2). In a 12-key keypad, letters 'a-z' are distributed among the keys 2 to 9. Similarly, 'space' is assigned to key 0. Since the numbers of keys are less than the English alphabets, each key is assigned with three or more letters.

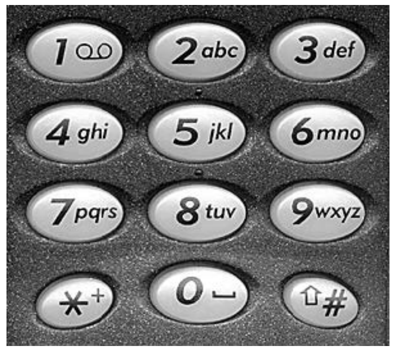

Figure 2: The standard 12-key keypad

(Silfverberg, MacKenzie, & Korhonen, 2000)

Several text entry techniques such as multi-tap and T9 follow the 12-key standard layout for text entry, which is briefly explained in next section.

2.1.1 Multi-tap

Multi-tap is a direct text entry method where each character is mapped by the user with no support from linguistic or dictionary database (Guha, 2009). The number of key presses depends on the position of the letter in the key. For instance, in a 12- key keypad key '2' has three letters on it which are 'a', 'b' and 'c'. Therefore, to enter 'a', press the key '2' once, for 'b' twice, for 'c' thrice and so on. In Multi-tap, awareness of the key position and key arrangement is easily formed as the characters are in alphabetical order, hence making text entry easier to learn. However, inserting characters from the same key in consecutively is somewhat cumbersome. For instance, if you are transcribing the word CAB, you press key '2' three times which will transcribe 'c', but then to type 'a' which is also in the same key '2', you will have to wait a few seconds before continuing. This wait causes a delay and affects the speed of text entry. This is called the 'time-out' problem. To remove this delay from the text entry method, you can also use a special key as a time-out kill. However, in general, multi-tap is not considered to be the most user-friendly text entry method (Nesbat, 2003).

According to Mackenzie (2002), the average number of keystrokes per character (KSPC) for multi-tap is 2.03.

2.1.2 Two-key method

To reduce the number of key press per character, another method called *Two-key method* was developed which used the same keypad form as multi-tap, but text entry was different as shown in the following section.

As the name suggests, the two-key method defines the use of two keys for text entry. The first tap selects the desired key where the character is present and the second key specifies the position of the character in that key. The position of the character ranges from one to four, as there are a maximum of four characters in a single key. For instance, if you have to type the letter ‘c’, press key ‘2’ first because key ‘2’ has the letter ‘a’, ‘b’ and ‘c’ inscribed on it, then press key ‘3’ which is the position of the letter ‘c’ in key ‘2’. Therefore, by pressing key ‘2’ and key ‘3’, the letter ‘c’ is entered in the text (second key, third position). This method requires less tapping of the keys than multi-tap (Butts & Cockburn, 2002).

2.1.3 LetterEase

LetterEase is an improvised version of multi-tap that eases and improves text entry in mobile phones via letter reassignment (Ryu, Cruz, & Zealand, 2005). The main motive behind the development of LetterEase was to improve multi-tap for text entry by overcoming its weakness of letter arrangement. In this arrangement, unlike multi-tap mapping three to four letters in a single key, two to three character per key were used utilizing keys 0, * and # as shown in Figure 3.

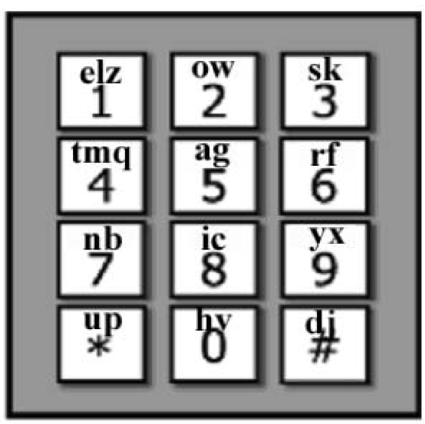

Figure 3: LetterEase keyboard layout

(Ryu et al., 2005)

The position of letters was selected by investigating the letter frequency, which determined the position and distance between the keys. Ryu et al., (2005) carried out a study with five male adult participants with a Nokia 3320 and Panasonic X60, where they found that the position of the thumb for the participant was around key '1'. Hence taking key '1' as the starting point, calculation on how far the thumb has to travel to type the letters was derived and based on that measure the position of the letters was assigned.

In this arrangement, the most frequently used letters are placed first in a key; with less frequently used letters in the keys second position. The position of the other letters from key '1' was also of concern so that the user does not have to move their thumb far from key '1'.

From the result of this experiment carried out with 15 participants, LetterEase was favored over a two-key method in terms of words per minute (WPM). In terms of keystrokes per character (KSPC), it was considered to be better than the multi-tap method. LetterEase had 6.92 WPM in its last 10 trials whereas the two-key method had 5.5 WPM. Similarly, it had 1.32 KSPC whereas multi-tap had 2.034 KSPC.

2.1.4 T9

In spite of different letter arrangements and techniques, direct text entry methods still had low performance; hence researchers developed a dictionary-based text entry method called 'T9', which supports users by disambiguating words after key is pressed. This system has an additional feature of dictionary, which is used to suggest words while texting.

'Text on 9 keys' (T9) developed and patented by Tegic Communications Inc., Seattle, WA is considered to be one of the leading text entry method choices of mobile companies (Nuance, 2012). Although it uses the same keypad layout as the multi-tap method, it removes the problem of multiple tapping of keys since each key is pressed only once. With the use of an English dictionary database, the system initially tries to figure out the intended letter when a key is pressed. If there is more than one possible match at the end of the word, the system displays other possible options to the user. More options are searched by using the 'Next' key, and when the search is completed, '0' key (space key) is used for the termination of selection (see Figure 4). For instance, if you want to type the word 'for', the key sequence 3, 6, 7 is pressed.

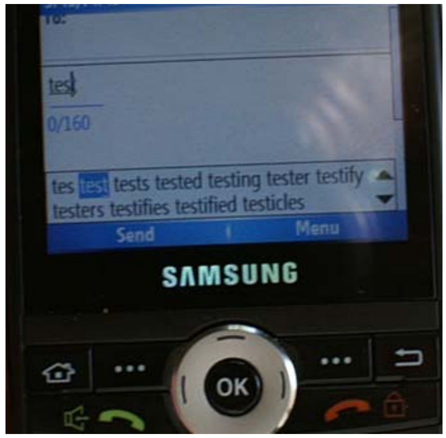

Figure 4: Text input with T9
(Shimpi, 2007)

However, the dictionary disambiguation, does not always work perfectly. Users have to press the next key and search for the right word. This process becomes a challenge to the user as their attention is focused towards the screen along with the keypad.

The other common problem with T9 is that not all words might be in the dictionary. In this case, user will prefer using the multi-tap method to T9 (Nuance, 2012).

An experimental study was carried out with twenty participants (novice and expert users) that compared two different methods of text entry, multi-tap and T9 (James & Reischel, 2001). From the study, the text entry rate for multi-tap users was 7.98 and 7.93 (novice and expert users) and for T9 users it was 9.09 and 20.36 (novice and expert users). It clearly indicates that text entry is faster using T9. A recent study comparing multi-tap and T9 text entry method with sixteen novice and experienced participants showed that a word per minute for multi-tap users was 10.93 and 11.29 (novice and expert users) and for T9 users 14.12 and 22.68 (novice and expert users) (Sajida, 2012). The results showed that experienced users are faster with text entry than novice users and the text entry is faster with T9. The result is similar with the study by James with a confirmation that in a gap of eleven years experience of users in using T9 has been increased.

2.1.5 iTap

Similar to T9, iTap is a predictive text entry method developed by Motorola. iTap suggests a word only when the user enters at least three letters in a row. Suggestions are based on the user's previous input and the word's frequency of occurrence. In this method, the right arrow key is used to select words other than the predicted one. For the final selection the 'up' arrow key of the keypad is pressed (see Figure 5).

Figure 5: iTap in action
(Wikipedia, 2012b)

As it is similar to T9, its features are also quite similar; however, T9 has been more widely accepted by mobile companies. The problem with dictionary-based systems is that the ambiguity increases as the user tries to type a non-dictionary word or some slang words or abbreviations. In such cases, user has to switch to multi-tap method where they have to type all the words that are not supported by the dictionary of the system. To remove this problem, a different technique called 'LetterWise' was developed based on prefix-based disambiguation rather than dictionary based (MacKenzie, Kober, Smith, Jones, & Skepner, 2001).

2.1.6 LetterWise

This predictive text entry method, developed by Eatoni, is based on the prefix based prediction method so that the system is able to predict the user's intended word that is beyond dictionary limitations. The letters preceding the current keystroke are called a 'prefix' upon which the anticipated letters are forecasted. For instance, if the user wants to type the word 'the', then he/she will press key '3' with the prefix 'th'. In this case, the preferred next letter will be 'e'.

In rare cases where LetterWise fails to predict the right character, the 'Next' key can be used to select other possible characters. This method aims at reducing the number of keystrokes per character while performing text entry.

In a longitudinal experiment carried out by (MacKenzie et al., 2001), LetterWise was compared with multi-tap with the help of twenty participants (ten multi-tap and ten LetterWise users) using a PC concepts KB-5640 numeric keypad (see Figure 6).

Figure 6: Keypad used in LetterWise Experiment
(MacKenzie et al., 2001)

From the experiment it was concluded that KSPC for LetterWise was 1.1500 and for multi-tap was 2.0342, thus confirming that LetterWise requires 43.5% less KSPC than multi-tap. Text entry speed for LetterWise was 36% faster than multi-tap. Following the concepts of LetterWise, another predictive text entry technique for hand held devices was developed called WordWise.

2.1.7 WordWise

This predictive text entry method, developed by Eatoni ergonomics, used a word-based mechanism, allowing easy and quick entry of name, addresses and URLs (Eatoni, 2007). It helps in predicting the whole word users would enter. WordWise works on the principle where the first letter you want to type is pressed only once. Upon further need of prediction, the second key that has the second letter of the word you want to type is entered. In this sequential manner, keys are pressed and the word you want to type appears on the screen. The presence of the 'Next Word' option allows you to choose among the list of suggested words. In case WordWise fails to predict your desired word, it can be switched to dictionary mode that activates the LetterWise method. After completing, the user needs to press 'space' key to get back to the 'WordWise' method (see Figure 7).

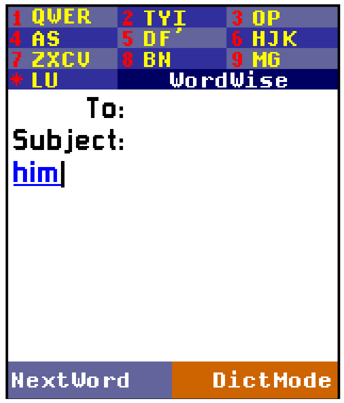

Figure 7: Wordwise in action

(Eatoni, 2007)

WordWise was later reformed to *Shift-WordWise*, designed basically for two thumb entries. Since a letter is entered holding the 'shift' key (key '1' as the shift key), it is known as 'Shift-WordWise'. Upon holding the shift key, the letters that will be entered are 'c', 'e', 'h', 'l', 'n', 's', 't' and 'y'. If key '1' or shift is not held while performing text entry, other letters present in that key get transcribed (see Figure 8). For instance, holding key '1' and pressing key '2' will transcribe the letter 'c', whereas pressing key '2' alone will transcribe letter 'a' or 'b' based on the former letter.

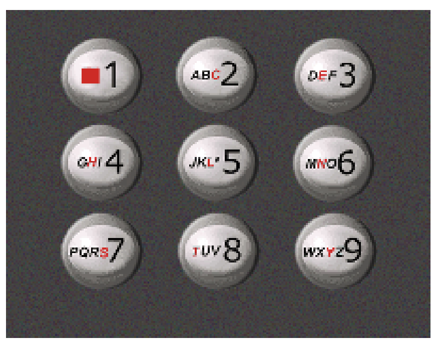

Figure 8: Shift-Wordwise keypad by Eatoni

(Eatoni, 2007)

Although invented, Shift-WordWise is still at a preliminary phase and is not commercially available. Hence, it is difficult to predict its commercial success. Other text entry methods, different than prediction methods, such as Fastap, Twiddler typing, TiltText were also developed, however, it was the QWERTY layout that was able to gain huge commercial success.

2.2 QWERTY Layout

With the evolution from typewriter to keyboard and from keyboard to mobile phones keyboard, QWERTY is one of the common text entry techniques in smart phones.

QWERTY layout was originally invented by Christopher Latham Sholes, a newspaper editor in early 1870's for a typewriter (Wikipedia, 2012c). It was then sold to E. Remington and Sons, who rearranged few keys and eventually the present QWERTY layout was originated (Yasuoka & Yasukoa, 2011). The first computer terminal to adopt QWERTY layout was 'Teletype' and it became the most common method with its acceptance to a QWERTY keyboard by Windows computers. It is believed that Nokia 9000 Communicator was the first mobile phone to feature full-QWERTY keyboard in 1996 (see Figure 9) (Kovach, 2011).

Figure 9: Nokia 9000 Communicator with full-QWERTY keypad
(Kovach, 2011)

Three conventional uses of the QWERTY layout in text entry are desktop keyboards, mini-keyboards for mobile devices, and soft keyboards used on touch-screens.

Due to the familiarity of arrangement of letters QWERTY layout was widely accepted in mobile phones. However, text entry rate in mobile phones is less compared to computers, which could be due to the limited space for an onscreen- keypad resulting in small key size. To remove the space adjustment problem, several mobiles allow text entry in landscape mode. Since holding a mobile phone in horizontal position allows more space for keys, some mobile phone models also implements landscape placement of keys (see Figure 10).

Figure 10: Samsung impression mobile phone with slide out landscape keypad
(GearCulture, 2012)

Several other reasons resulting the slow text entry rate using QWERTY in mobile phones includes:

- In the early years, touch sensors did not operate at high frequency, enough to detect very short touches in the keys.
- Early touch screens were capable of detecting only one touch at a time. Under these conditions touch-typing is difficult. However, new multi-touch screens are better in this area.
- Touch-typing is easy only in the size range of keyboards that fits the human hands. Both too small and too large keyboards are slower to operate.

2.2.1 Half-QWERTY

A modification of the QWERTY keyboard layout, the half-QWERTY keyboard layout (Matias, Mackenzie, & Buxton, 1994) uses two letters per key (see Figure 11). Therefore, the number of keys is halved resulting in the increase of operating modes. In this technique, a modifier key or space bar is held to type the text of the mirror - side keys.

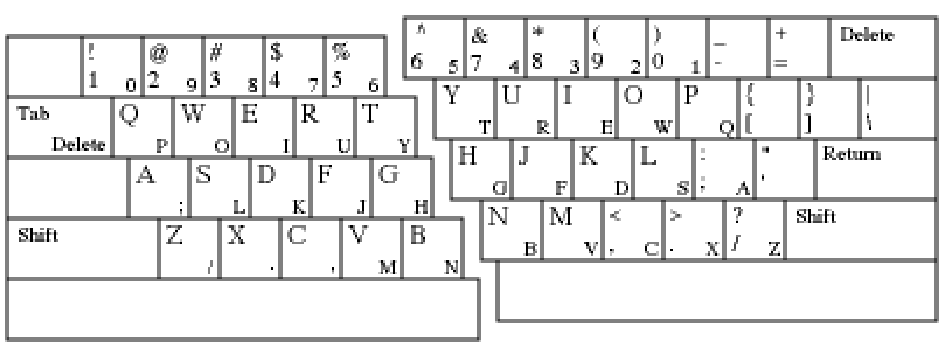

Figure 11: Left and Right handed Half-QWERTY layout on a standard QWERTY layout
(Matias et al., 1994)

Half-QWERTY keyboard size was resized to be used with only one hand. Although it has an advantage over two-handed full-QWERTY keyboard on its size, to increase the performance of text entry rate adequate practice is needed. Half QWERTY keyboard is comprised in a way where all the keys can be used single handed (Matias, Mackenzie, & Buxton, 1993). Half-QWERTY keyboard can be reduced in size thus making it easier to hold in the dominant hand (Matias et al., 1993). As a wearable computer, keyboard of Half-QWERTY is worn in one hand whereas the display screen can be placed on the other hand. This makes the system portable.

In an experiment conducted with 10 right handed participants to test the skill transfer from QWERTY to one handed keyboard using left hand for text entry, after 10 hours of learning experience, text entry speed of users ranged from 23.8 to 42.8 WPM, with 41% and 73% of their two handed text entry speed respectively (Matias et al., 1993).

2.2.2 Stick keyboard

The concept of stick keyboard originated from the fact that QWERTY acquires half of the screen of mobile device when transcribing text. Hence a reduced QWERTY keyboard that maps four rows of a standard QWERTY keyboard onto a single line was developed (Green, Kruger, Faldu, & St. Amant, 2004). This reduced QWERTY keyboard used 23% of the space of the original QWERTY keyboard and is known as 'Stick Keyboard' (see Figure 12).

Figure 12: Stick Keyboard layout

(Green et al., 2004)

In the stick keyboard, different characters are mapped into a single key. Keys are encoded via modifier keys or multi-tap input. Designers of stick keyboard were concerned about three factors of text entry: physical size, speed and learning curve. To type uppercase letters, numbers and symbols, modifier keys are used in stick keyboard. Similarly, the colour coding in the keyboard layout (for instance, orange colour of 'Smart Type' key, blue colour of 'Prev/ Next' key, red colour of 'Symbol Lock' key, and so on) works in a relationship with the modifier keys and the character produced (Green et al., 2004).

In an experiment conducted with 10 participants, stick keyboard with multi-tap input resulted in 10.4 WPM whereas stick keyboard with lexicon-based input resulted in 22.5 WPM. During the experiment matching words from vocabulary was displayed on the screen where the Next and Previous keys were used to traverse between selections. In accordance with their normal typing speed, stick keyboard with multi-tap reached to 22.9% of the original typing speed and lexicon-based input resulted in 48.2% of text entry speed (Green et al., 2004).

2.3 Five-key Text Entry

Five-key text entry, also known as date stamp method uses an interface of four cursor keys (up, down, left and right) and an enter key for text entry (MacKenzie & Soukoreff, 2002). Characters in five-key text entry are selected by rotating through the character set. Two buttons are used to scroll through the alphabet and enter key selects the letter (see Figure 13). This method is also used for entering text into electronic musical instruments and some small devices such as pagers.

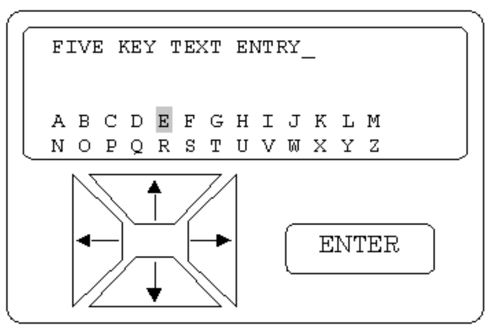

Figure 13: Five-key text entry method

(MacKenzie & Soukoreff, 2002)

Text entry rate using five-key text entry method is slow, since key press is required to move between characters. For decreasing the drawback of slower text entry rate, Bellman and MacKenzie (1998) purposed ‘Fluctuating Optimal Character Layout (FOCL)’, in which they arranged the letters, placing the most-likely used characters closer to the cursor position, thus minimizing the keystrokes per character by over 50%.

A study with 10 participants and 10 sessions, 15 minutes each was carried out to compare FOCL with five-key text entry method using QWERTY layout. No significant difference in the text entry rate was found between the two text entry methods. However, both FOCL and five-key text entry technique require much of user’s attention for scrolling and selecting characters. This is a responsible factor in decreasing the text entry rate.

2.4 SHK Card

SHK Card is a new input unit for mobile devices where users can input text using single hand: thumb to hold the card and the other four fingers to operate the keys (Sugimoto & Takahashi, 1996). In SHK Card, where keys are pressed to get the desired character, the control key (ambiguity resolution (AR)) key is pressed after the completion of text entry. There are eighteen keys arranged in 3×6 grid layout for single hand text entry (see Figure 14). Keys here are for text input and a stick and switches are for mouse data input.

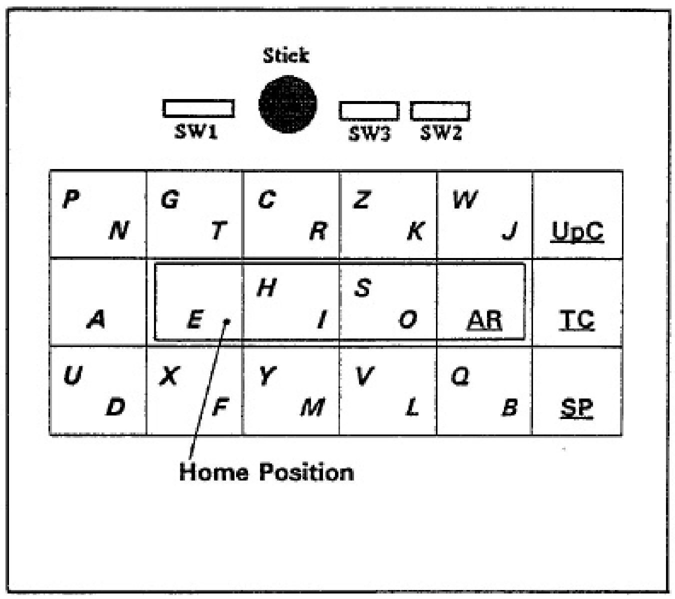

Figure 14: Keys, stick and switch arrangement on SHK Card
(Sugimoto & Takahashi, 1996)

SHK Card prototypes are made compatible with IBM PC compatible machines; however a concrete user testing for speed and accuracy is yet to be done.

SHK Card has not been implemented on a mobile phone; however, as they are approximately of the same size and shape as of mobile phone, there is a possibility of the implementation of SHK Card on a mobile phone.

2.5 TiltText

TiltText developed by (Wigdor & Balakrishnan, 2003), is a novel text entry technique which uses device orientation as a technique for text entry. Since 2003 tilt sensors have become a standard part of smart phones and therefore tilt-based interaction could be implemented without any additional parts using modern smart sphones. This technique can be used in standard 12 key mobile phones where by tilting the mobile phone in one of the four directions, a character is selected. It uses a combination of button press and tilting of the device to select a character which is quite similar to the technique designed by (Partridge & Chatterjee, 2002), called TiltType. For carrying out the tilt action a low cost sensor is incorporated within the mobile device. Keeping the phone untilted and pressing a key enters the numeric value present in that key. Similarly, left tilt is for entering first character, tilting the phone away from the user enters the second character; right tilt is for entering third character, whereas, fourth character is entered by tilting the phone towards the user (see Figure 15).

Figure 15: Text entry with TiltText

(Wigdor & Balakrishnan, 2003)

In an experiment conducted with 10 participants to compare the text entry speed of TiltText with multi-tap, it was found that with 5 hours of learning, TiltText was 23% faster than multi-tap (Wigdor & Balakrishnan, 2003).

2.6 Chorded Keyboard

As the name suggests, chorded keyboard works similar to the chord of a piano, pressing two or more keys together. It allows users to enter characters by pressing several keys together (Wikipedia, 2012d). Chorded keyboard can be designed to be use by one hand or two hands. For instance, *keyer* was an ergonomic chorded keyboard designed to be use without a board (see Figure 16).

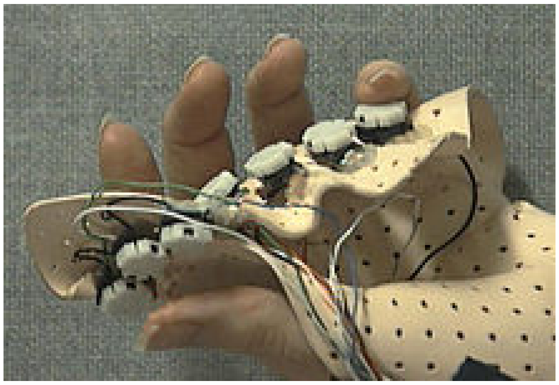

Figure 16: Chorded keyboard 'Keyer'
(Wikipedia, 2012e)

There have been several chorded keyboards developed, including *Microwriter* by Enfield Cie, USA in 1980, *Hall Braille Writer* by Frank Haven Hall in 1892 and The *Perkins Brailler* by David Abraham in 1951 (Wikipedia, 2012d).

Several studies of the chorded keyboards focus on its advantages over QWERTY, for instance, with the one handed chorded keyboard other hand is free. For similar reasons, a one handed chorded keyboard called ‘Twiddler’ was developed.

2.6.1 Twiddler typing

Twiddler is a popular one handed chording keypad having an average speed of 60 WPM for an experienced user(Lyons et al., 2004). It has a 3×4 layout grid, which is similar to a 12-button keypad (see Figure 17). While the user holds the device in his palm, four fingers are used to enter the text. Similarly, the thumb can be used to enter the special modifier buttons such as ‘Alt’, which is placed in the top near to users thumb.

Figure 17: Twiddler keypad
(Lyons et al., 2004)

An experiment with 10 participants and 20 sessions was conducted to compare the performance of twiddler typing. After the complete 20 sessions, users reached a speed of 26.2 WPM with twiddler, exceeding multi-tap's user speed of 19.8 WPM (Lyons et al., 2004).

3 Stylus-based Text Entry

In stylus-based text entry, texts are entered through tapping or gesture. In addition, user's finger is also used as a stylus for text entry. Stylus-based text entry involves soft keyboards, gesture recognition and handwriting recognition techniques.

3.1 Soft Keyboard

In a soft keyboard, a keyboard is drawn on a touch-sensitive display and is operated with finger or stylus tapping. Economically, manufacturing physical keyboard is costly. There are many parts that need to be correctly assembled in each key. With the virtual keyboard, this can be done in software. Furthermore, in soft keyboards, no physical keys are required; hence a mobile phone does not require dividing its space between the screen and keyboard. Keyboard appears in the screen only when text entry is a requirement. Different soft keyboard includes QWERTY soft keyboard, ABC soft keyboard, Quasi-QWERTY and OPTI.

With soft keyboards, texting is easier but on the contrary, more errors encountered while entering text. Lack of physical keys and the absence of tactile feedback resulted into more text entry errors. In iPhone, letters pop out when pressed making it easier for users to realize which key is pressed (see Figure 18). A different feedback mechanism was however needed to minimize text entry errors. An investigation on the use of vibro-tactile feedback in mobile devices (Brewster, Chohan, & Brown, 2007) suggests that the use of tactile feedback will result in improved interaction for users while texting with touch screen. According to the study, the performance of texting was improved when tactile effect was present with a key press.

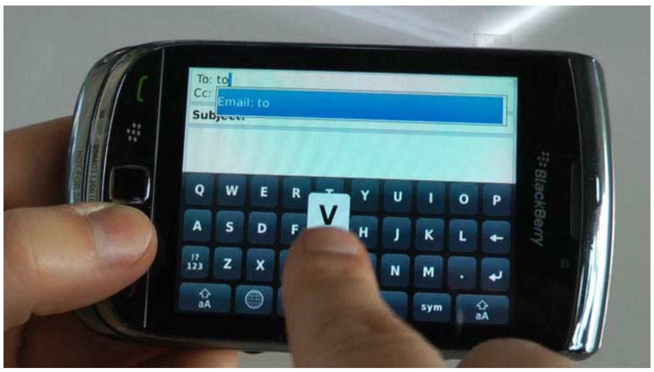

Figure 18: Blackberry Torch 9800 virtual QWERTY keypad
(UberMedia, 2012)

3.1.1 OPTI

OPTI keyboard layout was invented with a motive of designing an optimal soft keyboard which could be an alternative to QWERTY soft keyboard (see Figure 19) (Mackenzie & Zhang, 1999). While testing OPTI keyboard, a general comparison of text entry speed with QWERTY keyboard was performed in an experiment.

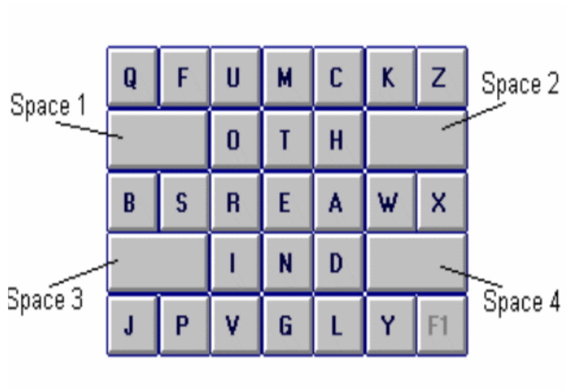

Figure 19: OPTI Keyboard Layout
(Mackenzie & Zhang, 1999)

OPTI key model follows several models.

1. Dictionary data – to have common pair of characters close to each other,
2. Fitts' law – to predict the time to tap a key given any previous key,
3. Shortest-path model – to compute the shortest path among several discrete paths and
4. Key-repeat-time measure – to calculate average key repeat time, was used.

A longitudinal experiment with 5 participants was carried out to compare the text entry rate of OPTI keyboard with QWERTY keyboard (Mackenzie & Zhang, 1999). The predicted maximum entry rate for OPTI was 58.2 WPM, which was about 35% faster than QWERTY. As expected, text entry rate for QWERTY was faster at the start of the experiment but a crossover of two techniques was also expected after few hours of practice. At the end of session 20, average entry rate for OPTI was increased from 17 WPM to 44.3 WPM. QWERTY started with high performance at the start of 28 WPM to 40 WPM till the last session. A crossover of the two techniques occurred in the 10th session, after a total of 4 hours of learning practice.

The results indicate that after a few hours of practice, OPTI is faster than QWERTY keyboard, thus making OPTI an effective alternative to QWERTY soft keyboard.

3.1.2 Quasi-QWERTY

Quasi-QWERTY is an optimized soft keyboard that has a layout similar to the standard QWERTY (see Figure 20). With the purpose of decreasing the visual search time of keys by the users, quasi-QWERTY utilizes the user's familiarity layout of the standard QWERTY layout (Bi, Smith, & Zhai, 2010).

q	w	d	r	t	u	y	l	k	p
z	a	s	e	h	n	i	o	m	
	x	f	v	c	g	b	j		

Figure 20: Quasi-QWERTY layout
(Bi et al., 2010)

Quasi-QWERTY layout has been optimized by the rearrangement of key position keeping consecutive letter pairs close to each other so that the time required for visual search of keys is minimized. Key rearrangement, however, follows the 3×10 ratio of QWERTY keyboard layout. For instance, key 'j' is moved to one step away from its original position, for instance to the position of key 'u', 'i', 'y', 'h', 'k', 'm' or 'n'.

An experiment was conducted with 12 participants where three keyboard layouts, Freely Optimized, Quasi-QWERTY and QWERTY were analysed (Bi et al., 2010). Results indicated that initial entry time for Quasi-QWERTY was mid-way between freely optimized keyboard and QWERTY keyboard, but the key arrangement of quasi-QWERTY was also helpful for the participants in locating keys easily than of the freely optimized keyboard layout.

3.2 Gesture-based Text Entry

Gestures different than other formal modes of communication, in general are a non-verbal communication method where a body's motion or posture is used for communicating messages. In the context of text entry, gesture based text input does not have a fixed set of strokes for character recognition; instead its framework interprets each pattern of users movement as a character (MacKenzie & Soukoreff, 2002). Different gesture-based text entry technique includes Cirrin, Quikwriting, T-Cube and Dasher, among which Cirrin and Quikwriting will be briefly studied in this chapter.

3.2.1 Cirrin

Cirrin is a gesture based text entry technique whose layout has a circular arrangement of letters and where users draw one stroke per word (Mankoff & Abowd, 1998). The letters are in a circular pattern to minimize the average distance between pairs of letter, so that while performing text entry, a long distance is not covered (see Figure 21). The presence of most likely consecutive characters, for example i, n, o, r letters, in a sequential pattern makes it easy to move the stylus to enter frequently used words in, or and on.

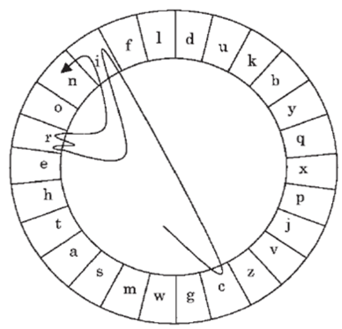

Figure 21: The 'Cirrin' interface
(Mankoff & Abowd, 1998)

In cirrin, a word-level unistroke soft keyboard is used which traces the path from the circumference of the circle. A word is entered without lifting the pen or stylus. There is no experimental evaluation done for its performance (Mankoff & Abowd, 1998), however, one user attained the speed of 20 WPM using cirrin layout text input on a daily basis for around two months. Similar to cirrin, another stylus based text entry technique called ‘Quikwriting’ is discussed briefly in next section.

3.2.2 Quikwriting

Quikwriting was developed with the motive of writing multiple texts continuously and fluidly (Perlin, 1998) so that there is no requirement to stop and lift the stylus for writing. Quikwriting interface is arranged in a 3×3 grid, numbered from 1 to 9 where 5th grid is the central resting zone. It has four sets of character modes for writing, which includes lowercase, uppercase; symbols and numeric mode (see Figure 22).

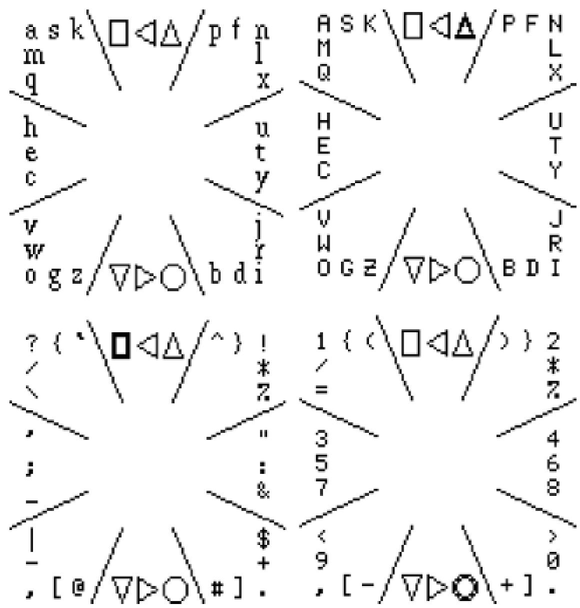

Figure 22: Four character sets of 'Quikwriting' interface
(Perlin, 1998)

For text entry in quikwriting, user has to start from the center zone 5, and move the stylus to different desired zones. When the intended character is selected, user has to return back to the resting (central) zone. For instance, to enter letter ‘k’, user starts from center zone 5, then move his/her stylus towards zone 1 where the letter ‘k’ is present, then continues to zone 2 and finally back to center zone. Similarly, for entering the characters that are in center of each zone, for instance, a, n, t, i, o, e, the stylus starts moving from zone 5 then to the chosen zone where the character is present and back to zone 5. These characters placed in the center of each zone are the most frequently used characters used in the English alphabet. According to a longitudinal study of 20 sessions with 12 participants (Isokoski & Raisamo, 2004), after the 20th session, Quikwriting reached 16 WPM which is 41% of QWERTY text entry rate (39 WPM).

The other gesture based text entry technique T-Cube (Venolia & Neiberg, 1994) is a technique similar to a two-tier pie menu system where each letter is selected using a flick gesture. Similarly, dasher is a technique that uses language modeling and continuous two-dimensional gesture to enter text (Ward, Blackwell, & MacKay, 2000). Dasher can be operated with one hand using a joystick, touch screen, trackball or mouse and even hands free by using an eye gaze tracker, which is useful for people with special needs (MacKay, 2011).

3.3 Word-gesture Keyboard

Conventional text entry system requires striking-based mechanism for text entry. Individual keys are tapped for writing text. In word-gesture keyboard, a continuous stroke gesture is done for text entry (Zhai & Kristensson, 2012). Letters in the keyboard are connected via a word-gesture mechanism, which approximately traces all the intended letters of the word. For typing the words that are out of dictionary individual letter keys are typed. If used frequently, the out of dictionary words will be added to the dictionary. Word-gesture keyboard is a good combination of soft keyboard and gesture based text entry. Till date, different text entry technique has implemented word-gesture mechanism, such as, ShapeWriter (see Figure 23), SlideIT, SHARK and Swype.

Figure 23:ShapeWriter (word-gesture keyboard)

(Zhai & Kristensson, 2012)

A notable advantage of gesture based text entry over tapping based text entry is the possibility of one continuous movement where lift is not required for typing each letter. This increases the speed of typing. As oppose to gesture based keyboard, in a word-gesture keyboard, user does not have to learn any gestures for text entry. Sliding the finger from one letter to another following the visual guidance to the next letter enters text.

3.4 Handwriting Recognition

Handwriting recognition is the ability of mobile devices to interpret the users' writing or scribbles to characters. The user uses a stylus to write on the screen surface, which is then read by the device as a character. Although handwriting is the most notable way of communication messages, handwriting recognition by devices is not so advanced. The obvious reason is because of the different ways of writing. For instance, each person has his unique way of writing such as distinct character writing or cursive writing, making it difficult to identify each character range, character extraction and recognition. The systems developed for handwriting recognition were an attempt to solve the problem of conventional handwriting.

Over the year's different handwriting recognizers such as 'Online recognizer', 'CalliGrapher', 'Apple-Newton Print Recognizer', 'ThinkWrite', 'Graffiti' and 'Unistrokes' have been developed. In this chapter, there is a brief discussion on Graffiti and Unistroke (see Figure 24).

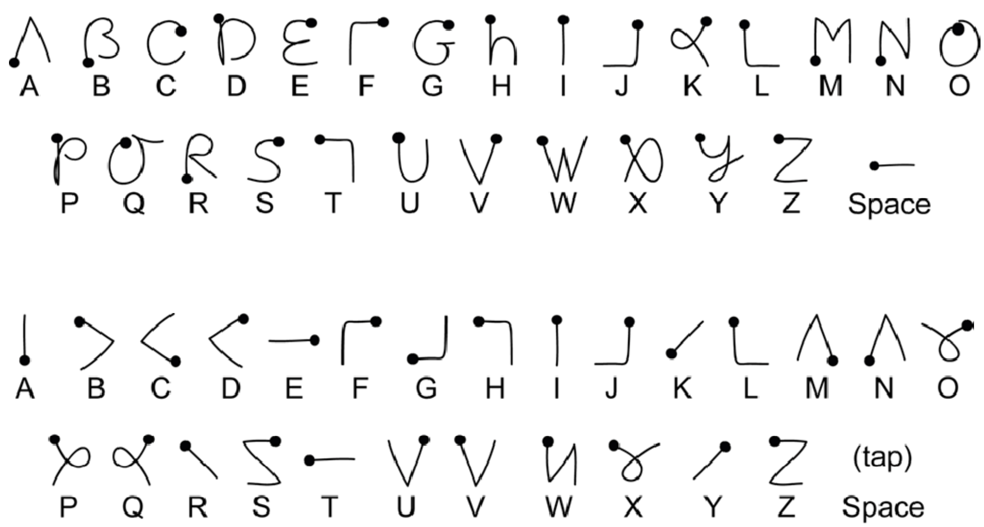

Figure 24: Graffiti (top) and Unistrokes (bottom) gesture alphabets
(Castellucci & Mackenzie, 2008)

Graffiti created and marketed by Palm, Inc. is a text entry system where a stylus is used to draw a stroke which resembles with the Roman letter assigned by the system, whereas another stylus based text entry, Unistrokes gestures do not resemble with the Roman letters (Castellucci & Mackenzie, 2008). Unistrokes recognizes all single stroke gesture alphabets, hence termed unistrokes. With Graffiti, users can enter letters, numbers and special characters by sliding their finger on the touch screen. Both the text input systems can be used in one handed form and have the potential to replace keyboards.

In an empirical comparison study of graffiti vs. unistroke, conducted with 10 participants over 20 sessions, it was found that the performance speed of Graffiti increased from 4 WPM to 11.4 WPM and for unistrokes from 4.1 WPM to 15.8 WPM, hence beating Graffiti in terms of performance speed. Both techniques however had high correction rates (Castellucci & Mackenzie, 2008).

3.5 MessagEase

Although several other text input technique were developed, none of them succeed in replacing QWERTY. According to Exideas Inc. this small key size of QWERTY keyboard is not suitable for text entry in touch screen devices (Exideas, 2012b). Hence they designed a unique technique of text entry, optimized for touch screens, that allow one finger or one handed full text entry (letters, numbers and special characters) in mobile phones, known as MessagEase (see Figure 25).

Figure 25: MessagEase keyboard layout in Android phone

(Brown, 2011)

Keyboard Concept

MessagEase is a novel and efficient text entry method which was primarily designed for cell phone using a 12 button keypad system, but with a different letter assignment technique (Nesbat, 2003). Although designed to replace QWERTY in soft touch screen devices, it is also applicable in PDAs, tablet computers and other such devices. MessagEase layout is based on letter frequency distribution such that the top 9 most frequent letters E, T, A, O, N, I, S of English alphabet are placed in 1-9 keys of the keyboard (Mayzner & Tresselt, 1965). This group of most frequently used letters comprises 71% of the text (Mayzner & Tresselt, 1965) and hence can be transcribed with a single tap. Similarly, the other group of 8 less frequently used letters V, L, X, M, F, W, Y, and K are allocated in two key sequences in such a way that they are closet to center key 5. Now the last group of less frequently used letters Q, U, P, B, J, D, G, and C are placed at the 8 directions of center key 5. The only remaining key Z is placed in key 8, with Z facing towards key 9. This is how letters are arranged in key sequence in a MessagEase keyboard (see Figure 26).

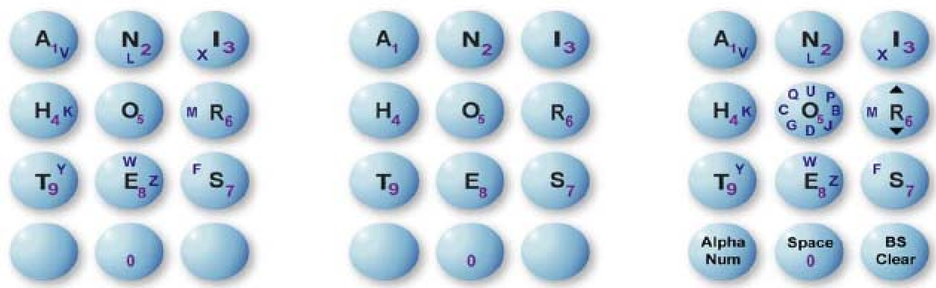

Figure 26: MessagEase letter arrangement

(Nesbat, 2003)

Text Entry with MessagEase

This unique letter arrangement of MessagEase has been designed to maximize the text entry speed with minimum errors. Due to its large key size, it is easier to text and the probability of tapping on wrong keys is less as compared to the other keyboard layouts with smaller key sizes. The basic technique for text entry with MessagEase is to tap for the most frequent letters, which appears at the center of each key with yellow color and drag or slide for the rest. You can tap anywhere on the key to enter those frequent characters. For instance, if you want to type letter A, tap anywhere at key 1 where letter A is present. Similarly to type V, start from anywhere of key 1 where letter V is inscribed and drag or slide to the direction where the desired letter is located, in this case towards the bottom right corner of key 1 where letter V is present. There is no boundary to stop for dragging, which makes dragging natural. Presence of visual and vibrational feedback makes the text entry quicker, easier and prevents from errors (see Figure 27).

Figure 27: Text entry with MessagEase (Tap and Drag feature)
(Googleplay, 2012)

In addition to the alphabets, numbers are transcribed through tap when the keyboard mode is changed to numeric mode by activating the 123 keys present at the right side of the keyboard. Similarly, special characters also appear when numeric mode is on and also in alphabets mode, which is typed by dragging as done with less frequent characters. With MessagEase, a new world record, Guinness World Record™ of entering 160 character phrase in 22.97 second has been noted which is equivalent to 84 WPM. It was achieved by Cheng Wei Chang of Singapore in September 4, 2012 (Exideas, 2012a).

MessagEase keyboard has a patent recognition, US Patent: 6,847,706 which endeavor for its principle of larger and fewer keys (Exideas, 2012c). MessagEase released its first version on October 07, 2010 and till November 20, 2012 it has released its 42nd version, Version 7.7.5 (Exideas, 2012d).The newer version have improved its gesture recognition, macros, word prediction, language support, fonts and wizards (Exideas, 2012d).

Since the layout of MessagEase is completely different from the usual QWERTY layout, which users are familiar with, users are required to put some effort in learning this technique for getting familiar with the different letters placement. MessagEase is currently available for free on Android, iOS, Windows and Pocket PC’s.

This study focuses on analyzing the learning curve of novice MessagEase users to figure out whether or not it can match up with the QWERTY performance.

4 Metrics of Evaluation

Text entry evaluation involves measuring speed and accuracy of different text entry techniques (Arif & Stuerzlinger, 2009; Soukoreff & Mackenzie, 2003). In general, for evaluating text entry techniques collection of phrases is used which the test participants enter. Entering a set of given phrases rather than creating phrases by the users is more efficient as the user does not require time to think on creating a phrase (MacKenzie & Soukoreff, 2002). This helps to measure the actual entry speed with minimum cognitive processing done by the user to think of phrases.

There are different approaches to experimental paradigms for evaluating text entry techniques. One of them does not allow incorrect letters by informing the users about incorrectly transcribed characters with the help of an audible beep(Isokoski & Käki, 2002; Venolia & Neiberg, 1994). Frequent occurrence of the audible beeps, however, might be frustrating for the users. A second approach allows every text entry disallowing corrections. In this case, a character once entered cannot be altered or corrected. A third approach is ignoring the errors encountered by the users while performing text entry (Lewis, LaLomia, & Kennedy, 1999; Tarvainen, 2010). A fourth approach involves allowing the users to transcribe text freely, quickly and accurately allowing correction of mistakes by using the backspace key. This unconstrained text entry evaluation paradigm analyzes data, thus being more informative (Wobbrock & Myers, 2006). I used the unconstrained paradigm in this study. In this chapter I will describe the metrics used in the evaluation of MessagEase.

4.1 Words per Minute (WPM)

Text entry speed was measured in words per minute (WPM) as follows:

$$WPM = \frac{|T|-1}{s} \times 60 \times 1/5 \text{(Wobbrock, 2007)}$$

where |T| is the length of the final transcribed string, ‘s’ is the time measured in seconds which starts with the entry of the first character and ends with the last character typed. Here, constant number 60 is the number of seconds per minute and the factor 1/5 denotes the average length of the words in character including spaces and numbers (Arif & Stuerzlinger, 2009).

4.2 Keystrokes per Character (KSPC)

Keystrokes per character (KSPC) are used to measure text input efficiency. It is the number of keystrokes required for generating a character of the text. KSPC is also applicable in devices where the text input is stylus based, such as strokes or taps (Mackenzie, 2002). Mathematically, KSPC is described as,

$$KSPC = \frac{|IS|}{|T|} \qquad \text{(MacKenzie \& Soukoreff, 2002)}$$

where 'input stream' is the actual keystrokes or character transcribed by the user that might include an incorrect character typed, or a backspace and 'transcribed text' is the final text transcribed by the user which is generally error free as the keystrokes that were used to correct the mistakes do not appear in the final transcribed text (Soukoreff & Mackenzie, 2003). The sign "|" denotes a function that computes the length of the string. Here, T is the actual string and |T| is the length of that string.

In QWERTY keyboard, error-free text entry has around 1.0 KSPC whereas in multi-tap phone keypad it is about 2.03 KSPC (Mackenzie, 2002).

4.3 Minimum String Distance (MSD) Error Rate

Total error rate is the addition of uncorrected error rate and corrected error rate, where keystrokes per character is the measure for corrected error rate and minimum string distance is the measure for uncorrected error rates. To solve the problem of matching incorrect transcribed input to the presented text, minimum string distance is calculated (Arif & Stuerzlinger, 2009). For instance, in any text entry experiment a set of text or phrases are presented to the user.MSD measures the minimum number of editing operations needed to make the strings identical i.e. if MSD>0, after the participant finished the strings were not identical (Soukoreff & Mackenzie, 2003). MSD Error rate is calculated as:

$$\text{MSD Error Rate} = \frac{\text{MSD (P,T)}}{\text{Max (|P|,|T|)}} \times 100\% \quad \text{(MacKenzie et al., 2001)}$$

where *P* and *T* are presented text and transcribed text strings, |.| represents length of the strings. Use of maximum length of presented and transcribed text in the denominator of the equation is to ensure that the error at upper limit is 100% (Soukoreff & Mackenzie, 2003).

4.4 Learning Curve

Every new technique, takes time for the users to get familiar with. New methods need to be practiced. Practice affects performance. The basic idea is with repetitive learning. Learning curve describes the development of performance over time.

The term 'learning curve' was first mentioned in 1936 by T.P. Wright, who created a mathematical model of learning curve for his studying time required to make aeroplane parts (Thomas, Mathews, & Ward, 1986). In text entry experiments, when a new technique is tested, user's performance is slow at the beginning users take a longer time to transcribe text; hence text entry rate is lower. With the increase of learning experience, WPM over time increases which is modelled by power law of learning (Card, Moran, & Lewell, 1983) as,

$$WPM = aX^b,$$

where a and b are regression coefficients and X is the variable of time.

5 Method

This chapter explains the methodology that was used to study beginner's learning performance using MessagEase. Chapter five includes four sections. First section deals with the initial hypothesis that was assumed before carrying out the experiment. It generally describes the goal that was set up prior to the experiment. In the second section, brief information about the participants of the experiment is given followed by information on the apparatus used during the experiment in the third section. In the fourth section, test procedure is explained. Finally, in the fifth section, study design is included.

5.1 Hypothesis

Beginner's performance with MessagEase is significantly lower than the performance of the same users on QWERTY soft keyboard. Based on the earlier research I expected low beginner performance for MessagEase, instead of the fact that experts can be very fast. With QWERTY keyboard, I expected increasing text entry rate with each session. The main goal of the experiment was to compare the feasibility of MessagEase with the current mobile text entry method, Virtual QWERTY. By comparing the learning experience of MessagEase users (beginner's) comparison of MessagEase text entry method was done with virtual QWERTY in terms of text entry speed and error rates. An experiment was carried to confirm the hypothesis.

The experimental evaluation also consists of the subjective evaluation of mobile users experience using MessagEase text entry method and virtual QWERTY text entry method.

5.2 Participants

10 participants (9 male, 1 female), aged 18 to 36 took part in a controlled experiment (see Figure 28). There were three sessions for each participant. Each participant typed 30 phrases per session, 15 phrases with MessagEase text entry method and 15 phrases with QWERTY method.

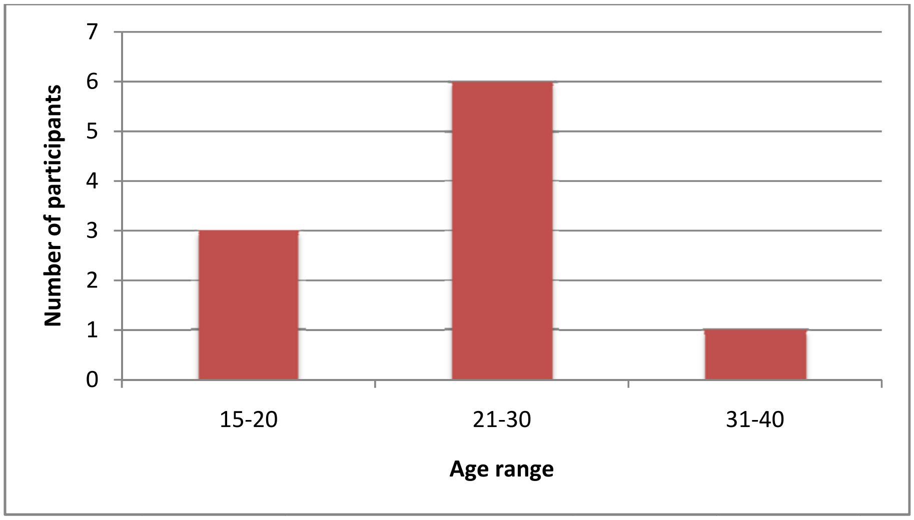

Figure 28: Age frequency distribution of participants of the experiment

Prior to the experiment, participants were asked to fill the background questionnaire form, where they were asked about their gender, age, familiarity with text entry method, text entry method in their mobile phone, amount of SMS they send per day, recent switch of text entry method if any and most familiar text entry method for them. Only one participant among all had recently switched the text entry method in mobile phone, form virtual QWERTY to physical QWERTY. All participants were familiar with text entry methods such as multi-tap, T9 or QWERTY and used to send SMS on a daily basis. Minimum number of SMS sent per day was of 0-1SMS per day for one participant whereas maximum was 30 SMS per day. Text entry method much used by the participants was virtual QWERTY, few used multi-tap too followed by T9 in few number. On the basis of the information collected, the number of message sent per day of each participant is shown in Figure 29.

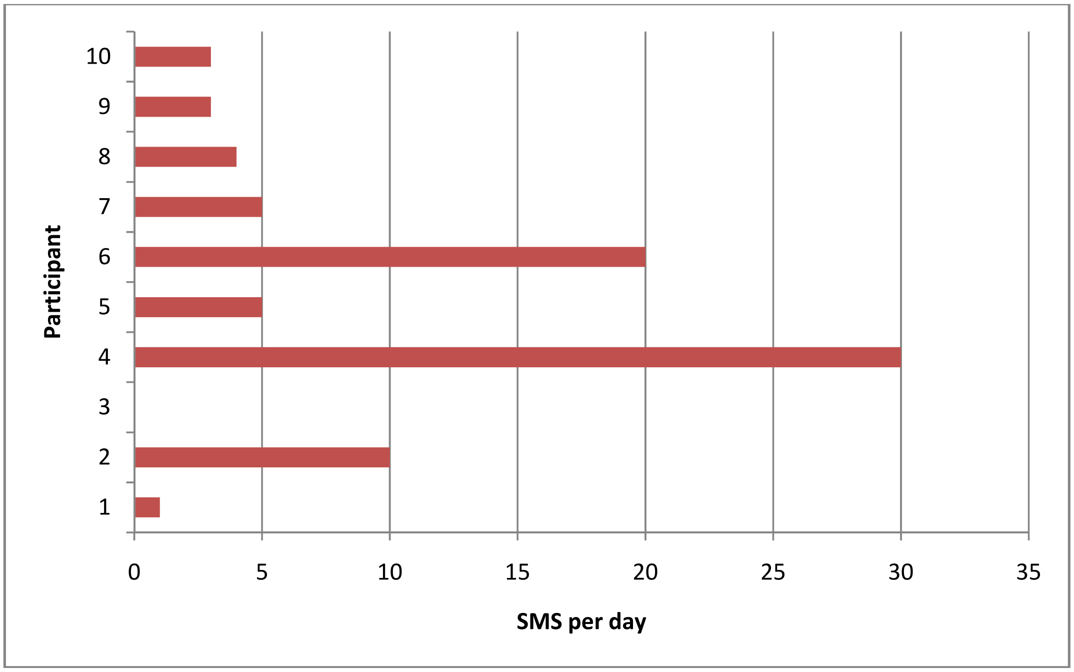

Figure 29: Number of SMS sent per day of each participant

None of the participants were familiar with MessagEase text entry method prior to the experiment, however most were experienced with virtual QWERTY layout.

5.3 Apparatus

In order to calculate text entry metrics, Text Entry Metrics on Android (TEMA) application was ran on ZTE Blade with Android OS v2.2 (see Figure 30). ZTE Blade is a smart phone with default virtual QWERTY keyboard. Throughout the experiment, the screen was held in portrait orientation; however the participants were free to hold the mobile phone in any position comfortable for them to transcribe text.

Figure 30: Mobile phone used in the experiment - ZTE BLADE

Software used in the experiment, TEMA is an application developed by Steven J. Castellucci and I.ScottMacKenzie of York University, Canada (see Figure 31) (Castellucci & MacKenzie, 2011).

Figure 31:Software used in the experiment (MessagEase text entry technique)

TEMA stores statistics log and event log. The start and end time of each text entry evaluation are recorded and event log contains time-stamped input events for post-study analysis. TEMA stored log file of presented text, transcribed text, presented character, transcribed character, input time in second, pause time in second, total time in second, words per minute, minimum string distance, number of backspace used, number of deleted characters, total error, corrected error and uncorrected error.Phrases to be transcribed during the experiment were selected randomly from the set of 500 phrases.

Using TEMA, two different text entry techniques were evaluated: Virtual QWERTY with dictionary prediction and MessagEase. QWERTY keyboard used in the experiment was the default text entry technique on the ZTE blade phone. Similarly, MessagEase version used in the experiment was v7.5.0, released on October 23, 2012. With both techniques, texts were to be entered in lowercase mode. As the auto-capitalization was on with MessagEase text entry method, participants switched the mode to lowercase while transcribing phrases. Input language used for text entry was standard English language.

Similarly, to transcribe a text, participant had to type the phrase shown in the ‘presented text’ in the application. All the letters were to be transcribed in lowercase mode and upon completing one phrase, user had to hit ‘Enter’ symbol for the next phrase to appear. In case of MessagEase text entry method, first letter from the first word of a sentence appeared in uppercase mode, hence users were asked to switch the mode to lowercase before beginning the transcription. When all 15 phrases were completed a notice of the completion of the experiment appeared on the screen.

5.4 Procedure

The experiment was divided into three sessions. In the first session, along with the pre-experiment questionnaire (see Appendix 1) and signing of the consent form, participants were introduced to MessagEase and QWERTY (see Appendix 2). Brief information about the methods, how to use and what needs to be done during the experiment was explained to the participants. First learning phase of the text entry methods was also performed during the first session. The first session lasted for about 40-60 minutes depending upon the text entry speed of the participants. The second session lasted 20-30 minutes. The third session included third learning phase along with the post experiment questionnaire and an informal interview (see Figure 32). In each session, participants transcribed 15 phrases with each text entry method.

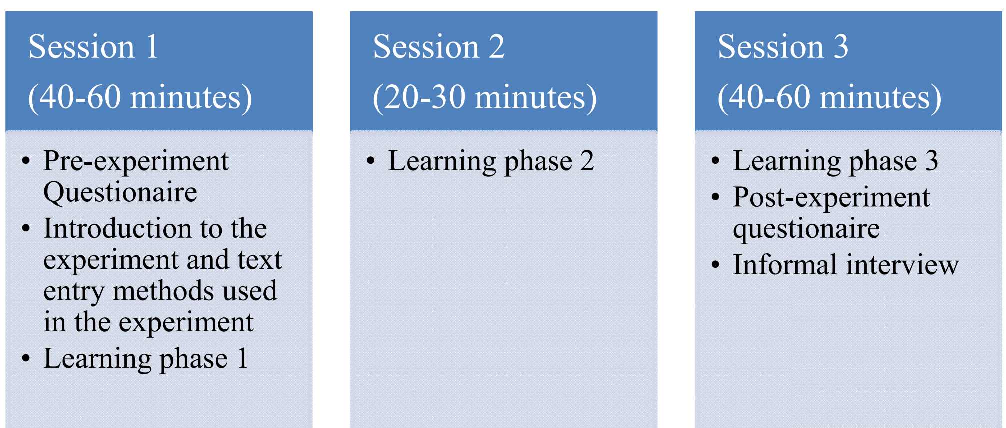

Figure 32: Experiment Schedule

The test was executed in the usability laboratory of the University of Tampere. In the first session of the experiment, participants were also informed about their right of leaving the experiment any time during the experiment if they feel uncomfortable. It was also highlighted that the testing is of the text entry method but not their skills. They were also asked to type as fast as possible and with minimum errors.

After the introduction of the experiment and the text entry methods used, it was made sure that the participants knew how to use the two text entry techniques. Written instructions were given (see Appendix 2) explaining the task, text entry methods and experiment goals. Along with the introduction during the first session, learning phase 1 was also completed. Participants were asked to type 15 phrases each, using both text entry techniques. To avoid learning associated with particular phrases, random phrases were selected. The order of the text entry techniques was switched for each participant. For example, if participant 1 starts the experiment with MessagEase in his/her first session, he/she was asked to start with QWERTY in the second session and again MessagEase in the third session. 5 participants started with MessagEase and the other 5 with QWERTY to counterbalance the order between participants.

During the second session, participants were asked to transcribe 15 sets of phrases using each text entry methods. In the third session, a third learning phrase was completed. Four examples of the phrases are shown in Table 1.

want to join us for lunch
is there any indication of this
the sun rises in the east
great disturbance in the force".

Table 1: Examples of phrases used in the experiment

After the end of the learning phase, participants were asked to fill the post-experiment questionnaire (see Appendix 3). The questions in it were based on the System Usability Scale (SUS) by John Brooke of Redhatch Consulting Ltd (Brooke, 1996).The SUS questionnaire is a simple ten question5-point usability scale that focuses on three main usability parameters- effectiveness, efficiency and satisfaction. Participants responded to the same questions regarding both text entry techniques. They were given the questionnaire form after the last trial of each text entry method.

Before wrapping up the experiment, each participant was asked about their experience with each technique, what they felt was good/bad about each text entry method used and what could be done to improve them.

5.5 Design

The experiment had a2×3 with-in subjects factorial design with text entry method (MessagEase, QWERTY soft keyboard) and amount of training (three sessions)as the independent variables.

The dependent variables were text entry speed and accuracy of transcribing text. Text entry speed was measured as words per minute (WPM).The accuracy was measured as the total error rate (TER), which is the sum of uncorrected error rate (UER) and corrected error rate (CER). Uncorrected errors are the errors that are not corrected by the participants while transcribing text. Corrected errors are, as the name suggests, those errors that were corrected (Castellucci & MacKenzie, 2011).

6 Results

This chapter deals with the result of the experiment. In the first section I describe the result for text entry speed in terms of words per minute (WPM). Second section shows the results for accuracy. Finally, in the third section results of the post-experiment questionnaire are presented.

6.1 Text Entry Speed

The analysis of variance of text entry speed showed that there was a significant effect of text entry method of entry speed ($F_{1,19} = 47.140$, $p < 0.0001$). The session (i.e. amount of training) also had a significant effect on text entry speed ($F_{2,18} = 3.631$, $p = 0.047$). Similarly, there was also a significant method by session interaction in the text entry speed ($F_{2,18} = 10.286$, $p=0.001$). The results showed that QWERTY scored higher text entry speed compared to MessagEase, whereas in MessagEase a significant growth in entry speed was seen with each session and learning experience. The overall mean for words per minute suggests that QWERTY was 85% faster than MessagEase.

In the start of the experiment, text entry speed with MessagEase was lowering than the text entry speed with QWERTY. Minimum text entry speed for a MessagEase session was 3.61 WPM and the maximum text entry speed was 16.75 WPM. Similarly, with QWERTY, minimum text entry speed was 8.54 WPM and the maximum text entry speed was 24.65 WPM. The mean text entry speeds and the standard deviations (SD) for each session are shown in Table 1.

Text Entry Method	Session 1	Session 2	Session 3	Mean
MessagEase	7.43 (SD= 2.23)	9.51 (SD= 2.52)	10.96 (SD=10.96)	9.3
QWERTY	17.75 (SD= 5.55)	16.78 (SD=4.92)	17.16 (SD= 4.46)	17.23

Table 2: Mean text entry speed (WPM)and standard deviation (SD) for two text entry methods

Text entry speed of MessagEase increased with each session (see Figure 33). Improvement in text entry speed of MessagEase was 28% from session 1 to session 2, and 15.25% from session 2 to session 3.

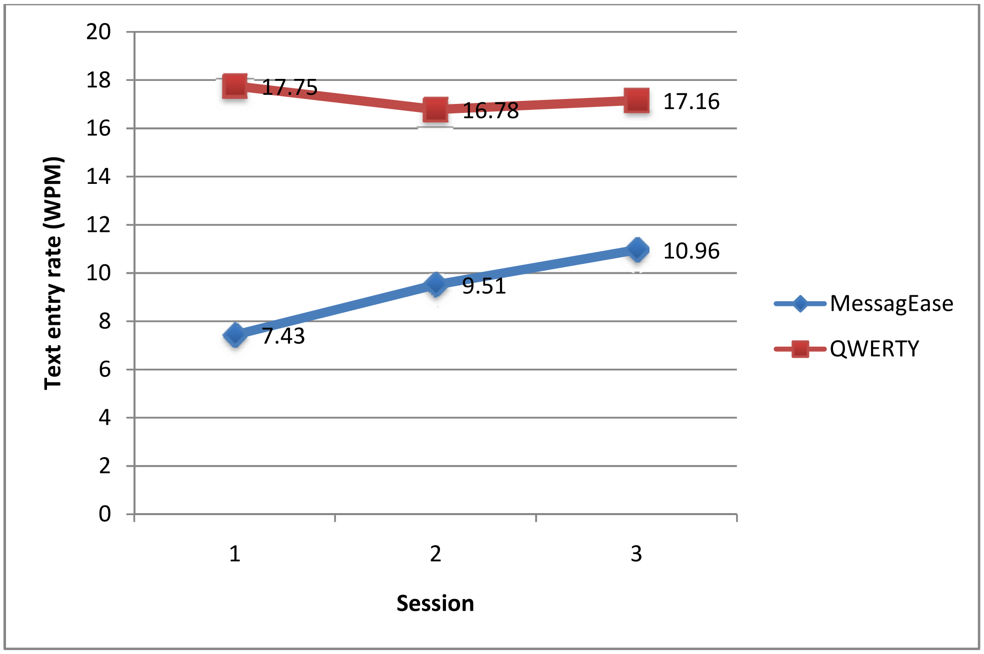

Figure 33: Text entry speed by entry methods and session

In opposition to what was expected, with QWERTY, a decrease of about 5% in the text entry speed was recorded from session 1 to session 2; following with an increase of 2.26% from session 2 to session 3 (see Table 1). However, the average text entry speed of MessagEase was 9.3 WPM. QWERTY scored 17.23 WPM in average.

For each text entry method, a regression model in the form of power law of learning was derived. This was done for analysing the performance levels of both text entry methods, if the experiment would have progressed. A learning curve of text entry methods to predict the performance according to the user behaviour has been illustrated up to 11th session (see Figure 34).

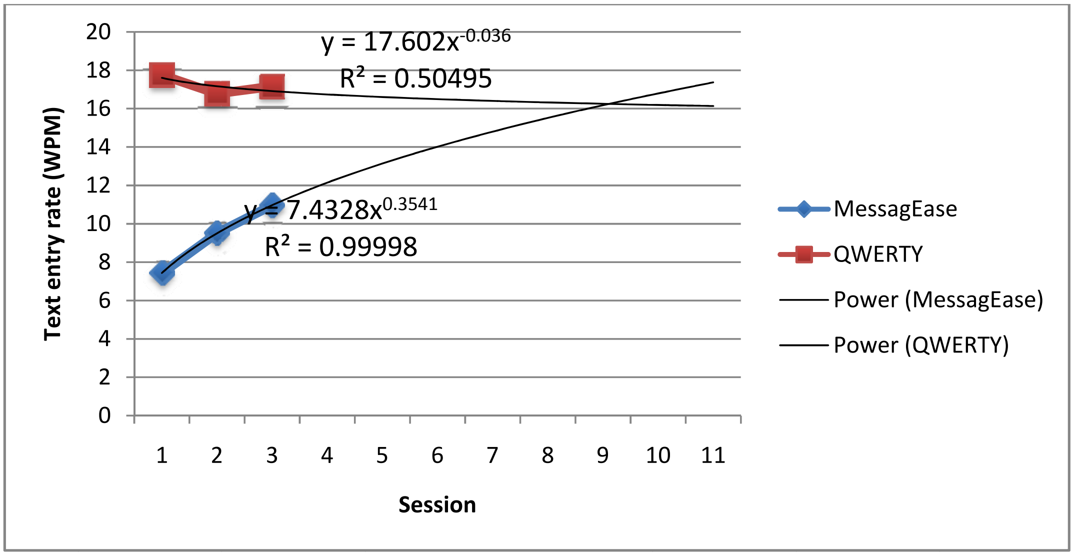

Figure 34: Learning curve behaviour to 11th session

According to the learning curve, the crossover would take place somewhere around the 9th session, after which MessagEase would surpass the performance of QWERTY layout. In this experiment, there are only three points, on the basis of which we cannot conclude the learning behaviour of text entry technique used.

6.2 Accuracy

For evaluating the accuracy of text entry method, total error rate was measured. Mean of error rate and standard deviation of MessagEase and QWERTY in all three session is shown in Table 2.

Text Entry Methods	Session 1	Session 2	Session 3	Mean
MessagEase	0.1037 (S.D=0.0569)	0.0922 (S.D=0.0351)	0.0752 (S.D=0.0307)	0.0904
QWERTY	0.0697 (S.D=0.0281)	0.0813 (S.D=0.0368)	0.0571 (S.D=0.0405)	0.0694

Table 3: Mean error rates and standard deviation (TER) for two text entry methods

Error rates of MessagEase decreased from 10% to 9% from first session to second session reaching to 7% on third session. Similarly for QWERTY it ranged from 7% to 6% from first session to third session. The analysis of variance of error rates between two text entry methods ($F_{1,9}$ = 4.486, p = 0.063) revealed no significant difference. There was also no significant difference in error rates in an interval of three sessions ($F_{2,18}$ = 2.097, p =0.152). Likewise, no method-session interaction was found ($F_{2,18}$ = 0.897, p = 0.425).

Although no statistically significant effects were observed we can try to interpret trends in Figure 35. Error rates for both text entry methods were similar and with no significant difference. The fact that there were no differences in error rate means that the differences in text entry rate are easy to interpret. Better text entry rate for QWERTY was not achieved at the cost of accuracy. On the contrary MessagEase seems to have caused slightly more errors.

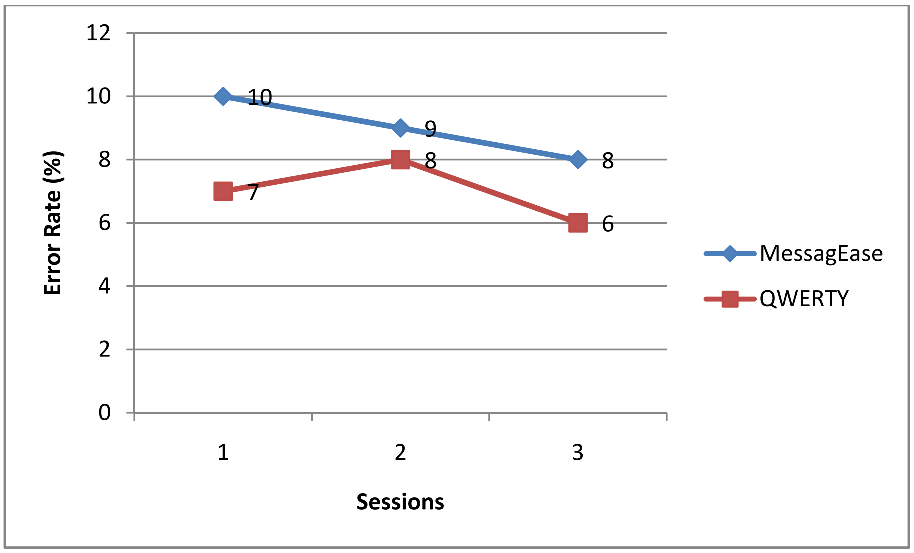

Figure 35: Error rates by text entry method and session

Total error rate using MessagEase layout continued to decrease from first session till the third session, however in QWERTY error rate increased from first session to second session.

6.3 Post-Experiment Questionnaire Result

In the post-experiment questionnaire the participants answered twenty post-experiment questions: ten of each text entry methods (see Appendix 3). The statements were made where the participants would agree or disagree based on System Usability Scale (SUS).

Participants rated the text entry techniques using a scale of 1 to 5, 1 = strongly disagree to 5 = strongly agree. The ratings of their agreement or disagreement with the statements were then analysed in the form of SUS score of each participants (see Figure 36). SUS scores ranged from 0 to 100. For calculating SUS score, first sum of scores from each statement was calculated. Here, for the statements that were positive about the system used, score equals to scale position minus 1. Similarly, for the rest statements, score equals 5 minus the scale position. 2.5 multiplied the final sum of score, which was the final SUS score for the text entry technique used.

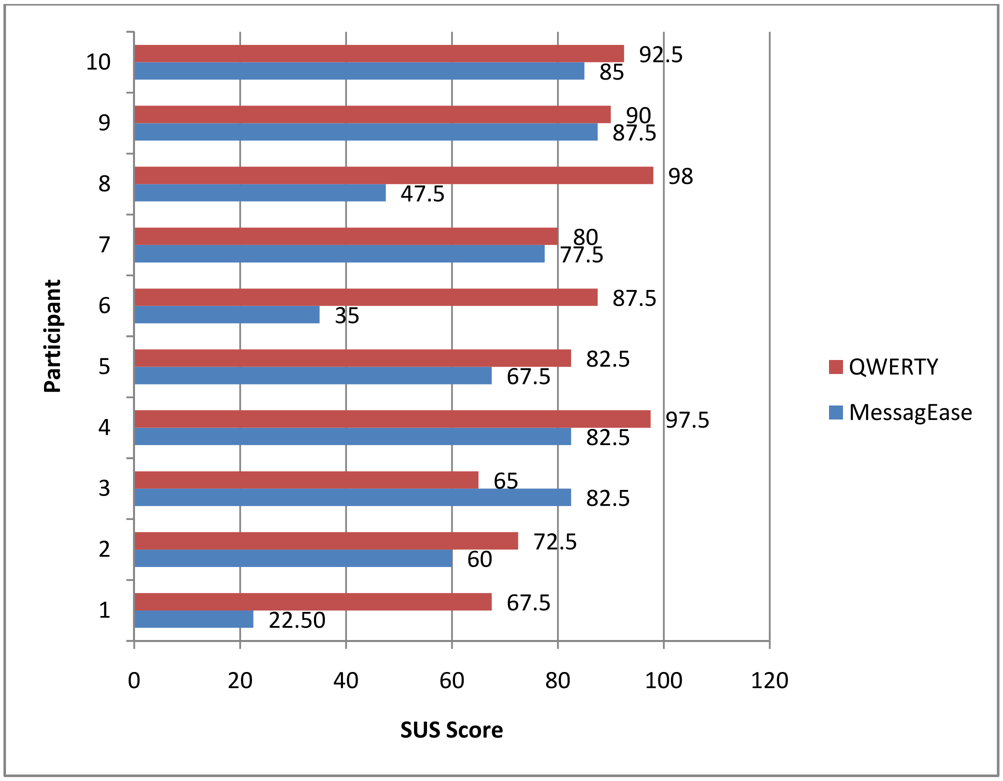

Figure 36: SUS Score by methods and participants

Mean of SUS score and standard deviation was considered (see Figure 37).

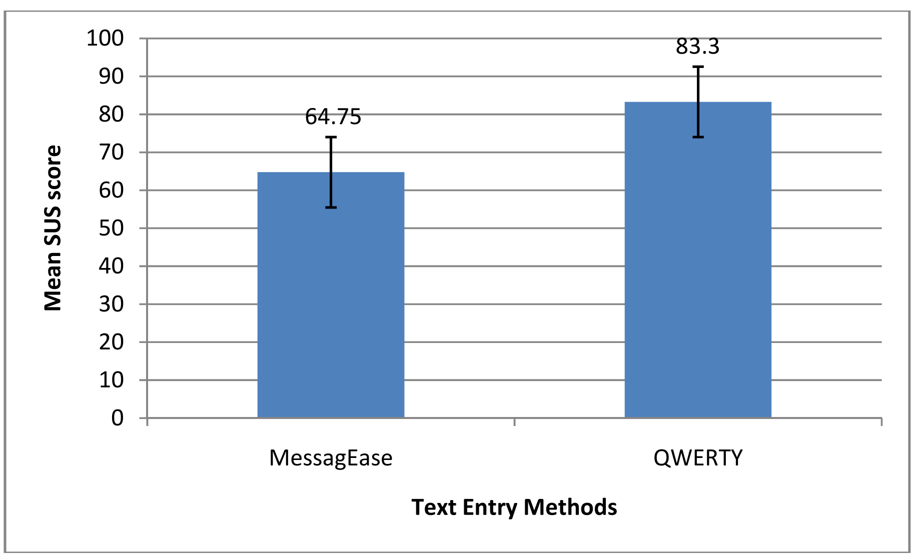

Figure 37: Mean SUS score by text entry methods

Minimum SUS score of MessagEase was recorded to be 23, which is less as compared to the minimum SUS score of QWERTY, i.e. 65. The maximum score of MessagEase was 88, whereas the maximum score of QWERTY was98, which is comparatively a high score. The comparison of average scores of MessagEase and QWERTY, 64.75 and 83.25 respectively clearly indicates that participants preferred QWERTY.

Statistically, the non-parametric Wilcoxon signed ranks test of SUS score showed that there was a significant difference in two text entry techniques ($z = -2.092$, $p = 0.036$).

7 Discussion and Conclusion

This thesis focused on the performance of MessagEase text entry method in comparison with QWERTY. A learning trend for both text entry methods was established to determine if and when a crossover point would occur. According to the measurements, new MessagEase users are very slow in comparison to their QWERTY performance.

The analysis of the result suggested that QWERTY has higher speed than MessagEase (see Table 1).This result was predictable. This experiment was intended for knowing the learnability and acceptance of a new text entry method, MessagEase. We preferred to have novice users of both text entry methods. However, QWERTY keyboard layout is not a new experience for computer and mobile phone users. An experienced QWERTY user is what we have in the real world.

In this experiment, we compared experienced use on QWERTY with novice use of MessagEase, which is not fair. MessagEase is actually comparing against QWERTY, which at this time everybody knows. Hence, MessagEase needs to demonstrate a good performance with user satisfaction under a condition where users are familiar with the performance of QWERTY. This experience might also be the cause of unusual learning curve of QWERTY. As shown in Figure 30, text entry speed of QWERTY is higher in session 1 as compared to session 2 and session 3. Participants might have been excited using the technique that they are experienced with during the first session resulting in better speed and fewer errors. Similarly, the learning curve of QWERTY and MessagEase method showed the crossover point at around Session 9 (see Figure 31), which is also somewhat unrealistic. If the learning curve of QWERTY were also progressive per session as of MessagEase, MessagEase would have required more learning hours to exceed QWERTY.

In addition, MessagEase layout was a new experience to all users. As in this experiment, participants tried both techniques in the same session, using MessagEase and QWERTY consecutively might have puzzled users with the layout.

However, there was no significant difference in the error rates of both techniques. Considering the fact that users were beginner with MessagEase but not QWERTY, we can conclude that with little more practice MessagEase would be more accurate than QWERTY. This may be due to its advantage of bigger key size.

The results from this experiment is similar to the study of comparing OPTI with QWERTY keyboard (Mackenzie & Zhang, 1999), where also QWERTY users were faster than OPTI users at the beginning of the experiment. According to the study, OPTI layout was poor initially with 17 WPM in 1st session, which reached to 45 WPM in 20th session. QWERTY layout ranged from 28 WPM to 40 WPM by 20th session. Based on the OPTI study we would have expected increasing performance for QWERTY also in our study, which however did not, happened. The crossover point of OPTI and QWERTY was at the 10th session which is somehow similar to our experiment.

In addition, error rates showed a different behaviour in two studies. In this study, error rate decreased over session but the difference was not statistically significant, whereas in the study by Mackenzie & Zhang (1999), error rate increased over sessions. Error rate for OPTI ranged from 2.07% to 4.18% and the error rate of QWERTY ranged from 3.21% to 4.84% from 1st session to 20th session. As per the result, QWERTY scored higher error rates throughout the experiment. Continuous increase in the error rates has been defined as the possibility of increase in speed over sessions that could have resulted in the increase of errors.

However, there are certain limitations in the study. This experiment was conducted in a controlled environment, which might not be the real environment where texting mobile actually occurs. External environment might vary from quiet environment to noisy places where people might be walking or eating food. Text entry experience might differ with environment; hence an experiment in real environment should also be considered during the study of text entry techniques. Besides, an expected distraction would be because of the use of new text entry technique where the user needs to concentrate on the ways to operate the new technique.

Similar variation to text entry might occur due to the language of the participants. Participants included in this experiment were not native English speakers (Isokoski & Linden, 2004); they might be more likely to send text in their native language. In this context too, experiment results might vary. Considering these factors while conducting an experiment could deliver better results.

A notable advantage of MessagEase over QWERTY, mentioned by every participants was on its keypad design with bigger space for keys. With smaller phone size, text entry becomes difficult if the key size is small. Hence, it suggests that MessagEase might be valuable especially in smaller phones. Spending few hours of learning could be useful for users with smaller phones. Users with mobile phones of bigger size may not find it useful to learn text entry with MessagEase. According to the user's, bigger space resulted in less mistake while entering text. However, unfamiliarity with the key combinations made the text entry difficult with MessagEase. As expected, QWERTY was favoured by the participants subjectively over MessagEase. MessagEase key combinations were hard to learn with few hours of practice, nevertheless all participants believed that with enough practice and learning experience MessagEase could have better performance. Creating a revolution in the trend is a big thing. Here, the main struggle with the adjustment of new text entry techniques is convincing the users. The main awareness that the new technology should consider in its development is that the technology should require very less effort to practice. If the system requires large amount of practice and is time consuming to get familiar with, it is hard to assure users with the system feasibility.

References:

Arif, A. S., & Stuerzlinger, W. (2009). Analysis of text entry performance metrics. *2009 IEEE Toronto International Conference Science and Technology for Humanity (TIC-STH)* (pp. 100–105). IEEE. doi:10.1109/TIC-STH.2009.5444533

Bi, X., Smith, B. a., & Zhai, S. (2010). Quasi-qwerty soft keyboard optimization. *Proceedings of the 28th international conference on Human factors in computing systems - CHI '10* (p. 283). New York, New York, USA: ACM Press. doi:10.1145/1753326.1753367

Brewster, S., Chohan, F., & Brown, L. (2007). Tactile feedback for mobile interactions. *Proceedings of the SIGCHI conference on Human factors in computing systems - CHI '07* (p. 159). New York, New York, USA: ACM Press. doi:10.1145/1240624.1240649

Brooke, J. (1996). SUS-A quick and dirty usability scale. *Usability evaluation in industry*, *189*, 194.

Brown, D. (2011). Essential Andoid Apps of the Wek. Retrieved December 17, 2012, from http://lifeonmymobile.com/2011/03/07/essential-android-apps-of-the-week/

Butts, L., & Cockburn, A. (2002). An evaluation of mobile phone text input methods. *Proceedings of the Third Australasian conference on User interfaces* (pp. 55–59). Darlinghurst, Australia: Australian Computer Society, Inc. doi:10.1145/563997.563993

Card, S. K., Moran, T. P., & Lewell, A. (1983). *The psychology of Human-Computer Interaction*. Lawrence Erlbaum Associates.

Castellucci, S. J., & Mackenzie, I. S. (2008). Graffiti vs . Unistrokes : An Empirical Comparison. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 305–308). New York, NY, USA: ACM. doi:10.1145/1357054.1357106

Castellucci, S. J., & MacKenzie, I. S. (2011). Gathering Text Entry Metrics on Android Devices. *CHI '11 Extended Abstracts on Human Factors in Computing Systems* (pp. 1507–1512). New York, NY, USA: ACM. doi:10.1145/1979742.1979799

Eatoni. (2007). Wordwise. Retrieved November 12, 2012, from http://www.eatoni.com/wiki/index.php/WordWise

Exideas. (2012a). MessagEase. Retrieved November 21, 2012, from http://www.exideas.com/ME/index.php

Exideas. (2012b). About Exideas. Retrieved November 21, 2012, from http://www.exideas.com/ME/about.php

Exideas. (2012c). Frequently Asked Questions About MessagEase. Retrieved November 21, 2012, from http://www.exideas.com/ME/faq.php

Exideas. (2012d). Version History. Retrieved November 21, 2012, from http://www.exideas.com/ME/downloadsAndroid.php

GearCulture. (2012). AT&T Samsung Impression. Retrieved December 17, 2012, from http://www.gearculture.com/men/gear/mobile-phones/page/3/

Googleplay. (2012). App Screenshots. Retrieved November 21, 2012, from https://play.google.com/store/apps/details?id=com.exideas.mekb&hl=en

Green, N., Kruger, J., Faldu, C., & St. Amant, R. (2004). A reduced QWERTY keyboard for mobile text entry. *Extended abstracts of the 2004 conference on Human factors and computing systems - CHI ’04* (pp. 1429–1432). New York, New York, USA: ACM Press. doi:10.1145/985921.986082

Guha, S. (2009). Text Entry Method for Reduced Keypads using One Key Stroke and One Column Stroke per Character. *13th International Conference on Human-Computer Interaction (HCI International 2009)* (pp. 76–80). San Diego: Springer. doi:10.1.1.164.6197

Isokoski, P. (2004). Performance of Menu-Augmented Soft Keyboards. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 423–430). New York, NY, USA: ACM. doi:10.1145/985692.985746

Isokoski, P., & Linden, T. (2004). Effect of foreign language on text transcription performance: Finns writing English. *Proceedings of the third Nordic conference on Human-computer interaction (NordiCHI ’04)* (pp. 109–112). New York, NY, USA: ACM. doi:10.1145/1028014.1028032

Isokoski, P., & Raisamo, R. (2004). Quikwriting as a multi-device text entry method. *Proceedings of the third Nordic conference on Human-computer interaction - NordiCHI ’04* (pp. 105–108). New York, USA: ACM Press. doi:10.1145/1028014.1028031

James, C. L., & Reischel, K. M. (2001). Text Input for Mobile Devices : Comparing Model Prediction to Actual Performance. *Proceedings of the ACM Conference on Human Factors in Computing Systems* (pp. 365–371). Seattle, WA, USA: ACM. doi:10.1145/365024.365300

Kovach, S. (2011). Here Are All The Smartphone Features You Love So Much And Who Had Them First. Retrieved November 16, 2012, from http://www.businessinsider.com/smartphone-firsts-2011-8?op=1

Lewis, J. R., LaLomia, M. J., & Kennedy, P. J. (1999). Evaluation of typing key layouts for stylus input. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, *43*(5), 420–424.

Lyons, K., Starner, T., Plaisted, D., Fusia, J., Lyons, A., Drew, A., & Looney, E. W. (2004). Twiddler Typing : One-Handed Chording Text Entry for Mobile Phones. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*

(Vol. 6, pp. 671–678). New York, NY, USA: ACM Press. doi:10.1145/985692.985777

MacKay, D. (2011). Dasher. Retrieved November 19, 2012, from http://www.inference.phy.cam.ac.uk/dasher/DasherSummary.html

Mackenzie, I. S. (2002). KSPC (Keystrokes per Character) as a Characteristic of Text Entry Techniques. *Proceedings of the Fourth International Symposium on Human-Computer Interaction with Mobile Devices* (pp. 195–210). Heidelberg, Germany: Springer-Verlag.

MacKenzie, I. S., Kober, H., Smith, D., Jones, T., & Skepner, E. (2001). LetterWise: prefix-based disambiguation for mobile text input. *Proceedings of the 14th annual ACM symposium on User interface software and technology* (Vol. 3, pp. 111–120). New York, NY, USA: ACM. doi:10.1145/502348.502365

MacKenzie, I. S., & Soukoreff, R. W. (2002). Text Entry for Mobile Computing: Models and Methods, Theory and Practice. *Human-Computer Interaction*, *17*(2), 147–198. doi:10.1207/S15327051HCI172&3_2

Mackenzie, I. S., & Zhang, S. X. (1999). The Design and Evaluation of a High-Performance Soft Keyboard. *Proceedings of the SIGCHI conference on Human Factors in Computing Systems* (pp. 25–31). New York, NY, USA: ACM. doi:10.1145/302979.302983

Mankoff, J., & Abowd, G. D. (1998). Cirrin : A word-level unistroke keyboard for pen input. *Proceedings of the 11th annual ACM symposium on User interface software and technology* (pp. 213–214). New York, NY, USA: ACM. doi:10.1145/288392.288611

Matias, E., Mackenzie, I. S., & Buxton, W. (1993). Skill Transfer From QWERTY. *Proceedings of the INTERACT ’93 and CHI '93 Conference on Human Factors in Computing Systems* (pp. 88–94). New York, NY, USA: ACM. doi:10.1145/169059.169097

Matias, E., Mackenzie, I. S., & Buxton, W. (1994). Half-QWERTY: Typing With One Hand Using Your Two-handed Skills. *Conference Companion on Human Factors in Computing Systems* (pp. 51–52). New York, NY, USA: ACM. doi:10.1145/259963.260024

Mayzner, M. S., & Tresselt, M. E. (1965). *Tables of single-letter and digram frequency counts for various word-length and letter position combinations* (pp. 13–32). Psychonomic Monographic Supplements.

Naughton, J. (2012). Now 4 billion people know the joy of txt. *theguardian*.

Nesbat, S. B. (2003). A System for Fast , Full-Text Entry for Small Electronic Devices. *ICMI’03 Proceedings of the 5th international conference on Multimodal interfaces* (pp. 4–11). New York, NY, USA: ACM. doi:10.1145/958432.958437

Nuance. (2012). T9. Retrieved November 5, 2012, from http://www.nuance.com/for-business/by-product/t9/index.htm

Partridge, K., & Chatterjee, S. (2002). TiltType : Accelerometer-Supported Text Entry for Very Small Devices. *UIST '02 Proceedings of the 15th annual ACM symposium on User interface software and technology* (Vol. 4, pp. 201–204). New York, NY, USA: ACM. doi:10.1145/571985.572013

Perlin, K. (1998). Quikwriting : Continuous Stylus-based Text Entry. *UIST '98 Proceedings of the 11th annual ACM symposium on User interface software and technology* (pp. 215–216). New York, NY, USA: ACM. doi:10.1145/288392.288613

Ryu, H., Cruz, K., & Zealand, N. (2005). LetterEase : IMPROVING TEXT ENTRY ON A HANDHELD DEVICE VIA LETTER. *OZCHI '05 Proceedings of the 17th Australia conference on Computer-Human Interaction: Citizens Online: Considerations for Today and the Future* (pp. 1–10). Australia: Computer-Human Interaction Special Interest Group (CHISIG) of Australia, Narrabundah, Australia.

Sajida, S. (2012). *New Data on Text Entry with Multitap and T9 Sajida Sajida*. University of Tampere.

Shimpi, A. L. (2007). Apple's iPhone: The Futue is Here. Retrieved November 10, 2009, from http://www.anandtech.com/show/2265/7

Silfverberg, M., MacKenzie, I. S., & Korhonen, P. (2000). Predicting text entry speed on mobile phones. *Proceedings of the SIGCHI conference on Human factors in computing systems - CHI '00* (Vol. 2, pp. 9–16). New York, New York, USA: ACM Press. doi:10.1145/332040.332044

Soukoreff, R. W., & Mackenzie, I. S. (2003). Metrics for text entry research : An evaluation of MSD and KSPC , and a new unified error metric. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 113–120). New York, NY, USA: ACM. doi:10.1145/642611.642632

Sugimoto, M., & Takahashi, K. (1996). SHK: Single Hand Key Card for Mobile. *Conference Companion on Human Factors in Computing Systems* (pp. 7–8). New York, NY, USA: ACM. doi:10.1145/257089.257093

Tarvainen, J. (2010). *Beginner Performance with the GKOS Chorded Keyboard Jussi Tarvainen*. University of Tampere.

Teixeira, T. (2010). Meet Marty Cooper - the inventor of the mobile phone. *BBC News*.

Thomas, H. R., Mathews, C. T., & Ward, J. G. (1986). Learning Cuve Models of Construction Productivity. *Journal of construction engineering and management 112(2)*, 245–258. doi:10.1061/(ASCE)0733-9364(1986)112:2(245)

UberMedia. (2012). Evolution of BlackBerry's Virtual Keyboard (Photos). Retrieved December 17, 2012, from http://n4bb.com/evolution-blackberrys-virtual-keyboard-photos/

Venolia, D., & Neiberg, F. (1994). T-Cube : A Fast , Self-Disclosing Pen-Based Alphabet. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 265–270). New York, NY, USA: ACM. doi:10.1145/191666.191761

Ward, D. J., Blackwell, A. F., & MacKay, D. J. C. (2000). Dasher---a data entry interface using continuous gestures and language models. *Proceedings of the 13th annual ACM symposium on User interface software and technology - UIST '00* (Vol. 2, pp. 129–137). New York, New York, USA: ACM Press. doi:10.1145/354401.354427

Wigdor, D., & Balakrishnan, R. (2003). TiltText : Using Tilt for Text Input to Mobile Phones. *UIST '03 Proceedings of the 16th annual ACM symposium on User interface software and technology* (Vol. 5, pp. 81–90). New York, NY, USA: ACM. doi:10.1145/964696.964705

Wikipedia. (2012a). Short Message Service. Retrieved December 17, 2012, from http://en.wikipedia.org/wiki/Short_Message_Service

Wikipedia. (2012b). iTap. Retrieved November 9, 2012, from http://en.wikipedia.org/wiki/ITap

Wikipedia. (2012c). QWERTY. Retrieved November 16, 2012, from http://en.wikipedia.org/wiki/QWERTY

Wikipedia. (2012d). Chorded keyboard. Retrieved November 22, 2012, from http://en.wikipedia.org/wiki/Chorded_keyboard

Wikipedia. (2012e). Chorded Keyboard. Retrieved November 12, 2012, from http://en.wikipedia.org/wiki/Chorded_keyboard

Wobbrock, J. O. (2007). *Measures of text entry performance*. (M. Kaufmann, Ed.) (1st ed., pp. 47–74). San Francisco, CA.

Wobbrock, J. O., & Myers, B. a. (2006). Analyzing the input stream for character- level errors in unconstrained text entry evaluations. *ACM Transactions on Computer-Human Interaction*, *13*(4), 458–489. doi:10.1145/1188816.1188819

Yasuoka, K., & Yasukoa, M. (2011). On the Prehistory of QWERTY. *Zinbun*, *42*, 161–174.

Zhai, S., & Kristensson, P. O. (2012). The Word-Gesture Keyboard : Reimagining Keyboard Interaction. *Communications of the ACM*, *55*(9), 91–101. doi:10.1145/2330667.2330689

Appendix 1

BACKGROUNDINFORMATION: BEGINNER'S PERFORMANCE WITH MESSAGEASE AND QWERTY

1. Name: __________________

2. Gender: Male [ ] Female [ ]

3. Age: __________________________

4. Familiar with text entry on mobile devices/ SMS: Yes [ ] No [ ]

5. Text entry method that you usually use with your mobile device:

- T9 [ ]
- Multitap [ ]
- Physical QWERTY [ ]
- Virtual QWERTY on a touch-screen [ ]
- Others, please specify____

6. How many SMS/email messages do you send with your mobile per day?_____________

7. Have you recently switched your text entry method? Yes [ ] No [ ]

If yes, please specify from _____________ to___________________

8. Most familiar text entry method: ____

Appendix 2

INTRODUCING MESSAGEASE and QWERTY KEYBOARD

MessagEase:

MessagEase is a text entry method for mobile phones and Tablet PCs. It is operated with one finger.

Fig 1: The MessagEase display

More Screenshots:

Fig 2: Tap feature of MessagEase

Instructions for using MessagEase:

1. Tap on the keys to enter the big characters “A NIH OR T E S”.
2. Tap and slide to enter other characters. The tap should land on the key where the desired character is displayed. The slide should go to the direction that the character is, in relation to the centre key.
3. For entering numbers tap on key 123. To switch back to the alphabet, tap again on the 123 key.
4. In the number mode special characters also appear. They are entered wit tapping and sliding just like the small characters in the alphabet display.
5. Alternatively, you can also have the special characters displayed in the main keyboard by dragging up from the SPACE key.
6. Dragging upon “r” button once, activates the shift key and dragging upon “r” button twice activates CapsLock key which changes the letters to uppercase until deactivated.
7. Dragging down from “r” button changes the following character from uppercase to lowercase and deactivates the CapsLock key if it is active.
8. Backspace arrow deletes one character when tapped. Tap and hold on the backspace arrow to delete multiple characters.

QWERTY:

Fig 3: The virtual QWERTY keyboard

Instructions for using QWERTY:

1. Characters are entered by tapping on the keys with the finger.
2. There will be a list of suggested words displayed above the keyboard. You can select word you want to type by tapping it with your finger.
3. To change the mode from uppercase to lowercase or vice-versa tap on the keys facing upwards. ⇧
4. For entering numbers and special characters tap on the 123 key present in the bottom left corner of the keyboard.
5. Back space key deletes one letter at a time.

Instructions for completing the task in the experiment:

1. Enter all phrases with lowercase characters. No numeric or special characters are needed.
2. With MessagEase, you need to change the keyboard from uppercase to lowercase by dragging down from "r" button before entering the first character. QWERTY will be in the lowercase mode automatically. You can select word you want to type by tapping it with your finger.
3. When you complete one phrase or sentence, press enter present at the bottom right corner of the displays of both text entry techniques.
4. Keep typing until a note that thanks you for your participation appears on the screen.

Appendix 3

EVALUATION FORM: BEGINNER'S PERFORMANCE WITH MESSAGEASE AND QWERTY

Please rate the text entry techniques as you experienced while using them. You have to rate using the scale of 1 to 5, 1= strongly disagree – 5 = strongly agree.

A) MessagEase

	Strongly disagree 1	2	3	4	Strongly agree 5
1. I think that I would like to use this system frequently					
2. I found the system unnecessarily complex					
3. I thought the system was easy to use					
4. I think that I would need the support of a technical person to be able to use this system					
5. I found the various functions in this system were well integrated					
6. I thought there was too much inconsistency in this system					
7. I would imagine that most people would learn to use this system very quickly					
8. I found the system very cumbersome to use					
9. I felt very confident using the system					
10. I needed to learn a lot of things before I could get going with this system					

B) QWERTY

	Strongly disagree 1	2	3	4	Strongly agree 5
1. I think that I would like to use this system frequently					
2. I found the system unnecessarily complex					
3. I thought the system was easy to use					
4. I think that I would need the support of a technical person to be able to use this system					
5. I found the various functions in this system were well integrated					
6. I thought there was too much inconsistency in this system					
7. I would imagine that most people would learn to use this system very quickly					
8. I found the system very cumbersome to use					
9. I felt very confident using the system					
10. I needed to learn a lot of things before I could get going with this system					